University of Kentucky
UKnowledge

University of Kentucky Doctoral Dissertations | Graduate School

2009

THREE-DIMENSIONAL FREE SURFACE NON-HYDROSTATIC MODELING OF PLUNGING WATER WITH TURBULENCE AND AIR ENTRAINED TRANSPORT

Tien Mun Yee
University of Kentucky, tmyee0@engr.uky.edu

Right click to open a feedback form in a new tab to let us know how this document benefits you.

Recommended Citation
Yee, Tien Mun, "THREE-DIMENSIONAL FREE SURFACE NON-HYDROSTATIC MODELING OF PLUNGING WATER WITH TURBULENCE AND AIR ENTRAINED TRANSPORT" (2009). *University of Kentucky Doctoral Dissertations*. 699.
https://uknowledge.uky.edu/gradschool_diss/699

This Dissertation is brought to you for free and open access by the Graduate School at UKnowledge. It has been accepted for inclusion in University of Kentucky Doctoral Dissertations by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.

ABSTRACT OF DISSERTATION

Tien Mun Yee

The Graduate School

University of Kentucky

2009

THREE-DIMENSIONAL FREE SURFACE NON-HYDROSTATIC MODELING OF PLUNGING WATER WITH TURBULENCE AND AIR ENTRAINED TRANSPORT

ABSTRACT OF DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy in the
College of Engineering
at the University of Kentucky

By
Tien Mun Yee

Lexington, Kentucky

Director: Dr. Scott A. Yost, Associate Professor of Civil Engineering

Lexington, Kentucky

2009

Copyright © Tien Mun Yee 2009

ABSTRACT OF DISSERTATION

THREE-DIMENSIONAL FREE SURFACE NON-HYDROSTATIC MODELING OF PLUNGING WATER WITH TURBULENCE AND AIR ENTRAINED TRANSPORT

The advance in computational fluid dynamics in recent years has provided the opportunity for many fluid dynamic problems to be analyzed numerically. One such problem concerns the modeling of plunging water into a still water body, often encountered in pump stations. Air bubbles introduced into the system by the plunging jet can be a significant problem, especially when consumed into operating pumps. The classical approach to investigate the hydrodynamics of plunging jet in pump stations is by physical model studies. This approach is time consuming, tedious and costly. The availability of computational power today, along with appropriate numerical techniques, allows such phenomenon to be studied in a greater level of detail and more cost efficient. Despite the advantages of numerical studies, little attention has been devoted to solve the plunging jet and air transport problem numerically.

In this current work, a *3*-dimensional finite volume, Large Eddy Simulation (LES) code is developed to simulate these flow conditions. For turbulent flow, the large scale quantities were numerically resolved while the dynamic sub-grid scale model is used to model the small scale energy dissipations. The code also has the capability to handle free surface deformation, an important aspect in simulating the impact section of an impinging jet.

Modeling of the air entrainment is performed numerically utilizing the information obtained from the hydrodynamics. Migration of air bubbles is modeled using the scalar transport equation, modified to account for the buoyancy of the bubbles. Instead of the typical Lagrangian schemes, which track individual air bubbles, air bubble dynamics are modeled in the form of concentrations. Modeling air bubbles in this manner is computational efficient and simpler to implement. For the air entrainment simulations, standard numerical boundaries conditions and empirical entrainment equations are used to provide the necessary boundary conditions. The developed model is compared with the literature, producing satisfactory results, suggesting that the code has an excellent potential of extending its application to practical industry practices.

KEYWORDS: Air bubbles transport, pump station modeling, overfall plunging jet, Finite volume method, Large Eddy Simulation.

Tien Mun Yee

05/06/2009

THREE-DIMENSIONAL FREE SURFACE NON-HYDROSTATIC MODELING OF PLUNGING WATER WITH TURBULENCE AND AIR ENTRAINED TRANSPORT

By

Tien Mun Yee

Scott A. Yost

Director of Dissertation

Kamyar C. Mahboub

Director of Graduate Studies

05/06/2009

RULES FOR THE USE OF DISSERTATIONS

Unpublished dissertations submitted for the Doctor's degree and deposited in the University of Kentucky Library are as a rule open for inspection, but are to be used only with due regard to the rights of the authors. Bibliographical references may be noted, but quotations or summaries of parts may be published only with the permission of the author, and with the usual scholarly acknowledgments.

Extensive copying or publication of the dissertation in whole or in part also requires the consent of the Dean of the Graduate School of the University of Kentucky.

A library that borrows this dissertation for use by its patrons is expected to secure the signature of each user.

Name	Date

DISSERTATION

Tien Mun Yee

The Graduate School

University of Kentucky

2009

THREE-DIMENSIONAL FREE SURFACE NON-HYDROSTATIC MODELING OF PLUNGING WATER WITH TURBULENCE AND AIR ENTRAINED TRANSPORT

DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy in the
College of Engineering
at the University of Kentucky

By
Tien Mun Yee

Lexington, Kentucky

Director: Dr. Scott A. Yost, Associate Professor of Civil Engineering

Lexington, Kentucky

2009

Copyright © Tien Mun Yee 2009

I would like to dedicate this dissertation to my family, especially…

my father, Soon Wah Yee,

my mother, Yee Soon Liew,

my wife, Feena Mei Ching Wong and

my son, Jae Ren Yee.

ACKNOWLEDGEMENTS

I am indebted to Dr. Scott A. Yost and Dr. Scott F. Bradford for providing me with unconditional advice, direction and guidance in the course of pursuing this work. Dr. Yost has been an excellent academic advisor and mentor in all the years I have worked for him. During some tough times, he had never once given up on me and has been working patiently, teaching and encouraging me in every aspect of my life. Dr. Yost had given me one of the most valuable things in life, my higher education. I would not and will not trade my experience working for him for anything else. He is without any doubt one of the most influential people in my life.

The role of Dr. Bradford had been most significant and is still an understatement. This work would not have been possible without his contributions. He is an inspiration and invaluable resource in the field of computational fluid dynamics (CFD). Dr. Bradford had taught me so much in CFD coding, as well as some important basics of CFD. I did not have the opportunity to meet with him personally, but all the email correspondence with Dr. Bradford have either directly or indirectly taught me more than what I could have learned in years. Dr. Bradford has sacrificed his time replying to my questions in a very promptly manner. Most unfortunately, due to university regulations, he could not officially serve as one of the committee member for this dissertation.

Dr. T.T.H. Tsang and Dr. J.M. McDonough had provided some much needed help in sharing their wealth of knowledge, time and patience in lengthy discussions on issues related to turbulence modeling. The discussions often lead to critical thoughts and

motivations to further improve my work, and more importantly, correct the mistakes I have made.

I would also like to acknowledge Dr. Jimmy Fox and Dr. Gail Brion, for their encouragements and advices during the course of this work and their service on my committee for this dissertation.

Furthermore, I would like to thank my family members, especially my wife for her unconditional love and support, my parents, my brother and sisters for their emotional, and at times, financial support throughout my long academic pursuit.

I would like to take this opportunity to thank my friends and colleagues King Fu Hii, Pang Leen Ong, Brian Belcher and Zhiyu Shao for their help, support and friendship. Their friendship made the journey to the end of this dissertation so much more enjoyable and memorable.

Last but not least, I would also like to thank KWRRI for funding part of this research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

a	Equivalent radius for air bubble
A, A^{-1}	Matrix and matrix inverse of a system of equations
A^{+}	Damping function constant *25*
$a_1, a_2, \cdots, a_9$	Coefficients for the A matrix in free surface equations
A_g	Cross section area for gaseous phase
ag_1, ag_2	Normalized grid length scales/grid ratio
α	Arbitrary vector or scalar
a_u, a_d, a_l	Tridiagonal matrix coefficients
A_{wave}	Amplitude of the solitary wave
$b_1, b_2, \cdots, b_{19}$	Coefficients for the A matrix in Pressure Poisson Equation
β	Beta family average coefficient
b_n	Digital filter coefficients
b_s	Thickness of jet
b_w	Upstream channel width
C	Dynamic model coefficient, C_s^2
C_α	Propagation velocity
C_d	Coefficient of drag
$C_{d\infty}$	Coefficient of drag at terminal velocity
C_{dw}	Wall coefficient of drag
χ	Impinging angle
C_s	Classical Smagorinsky coefficient
c_{wave}	Celerity or wave speed
D	Total water depth at a cell face
D_0	Still water depth

D_{brink}	Brink depth at upstream channel
$D_{critical}$	Critical depth at upstream channel
Δ	Temporal or spatial grid size/ LES grid filter width
δ	Channel half width
$\widetilde{\Delta}$	Test filter width
Δt	Time step
D_f	Damping function
D_g	Diameter of air bubble
D_T	Eddy diffusivity in scalar transport equation
e	Least squared error
Eo	Eötvös number
F_i, H_i, P_i	Vector quantities in generalized coordinate system
F_b	Buoyancy force of bubble
$\widetilde{\boldsymbol{f}}, \widetilde{\boldsymbol{g}}, \widetilde{\boldsymbol{h}}$	Vector quantities representation for the momentum equations
$fc_\xi, fc_\eta, fc_\zeta$	Advection and diffusion fluxes in scalar transport equation
F_d	Drag force by the bubble
f_f	Friction factor for air bubble
$fi(x, y)$	Vector or scalar quantities
F_{net}	Net force acting on the bubble
Fr	Froude number
F_{total}	Total force acting in a cell
f_x, f_y, f_z	Coriolis parameter in the three respective directions
g	Acceleration due to gravity
$G(x, y)$	Function for low pass filter
H	Free surface elevations
h_{fall}, H_0	Vertical drop height of overfall
h_{wave}	Wave profile with reference to still water depth

κ	Von Karmon constant, *0.41*
k_s	Roughness height
L_{ij}	Leonard stresses in tensor notation
l_{ij}	Grid metric terms in generalized coordinate
l_s	Width of jet
m_i	Vertical shear stress in generalized coordinate
M_{ij}	$-\tilde{\Delta}^2 \left\|\tilde{\tilde{S}}_{ij}\right\| \tilde{\tilde{S}}_{ij} + \Delta^2 \widetilde{\left\|\tilde{S}_{ij}\right\| \tilde{S}_{ij}}$
Mo	Morton number
μ	Dynamic viscosity
N	Number of air bubble
N_f	Spatial or temporal extent of filter
ν, ν_l	Molecular viscosity
ν_T	Turbulent eddy viscosity
$\mathcal{O}$	In the order of
p	Dynamic pressure
P	Total pressure
ϕ	Arbitrary Scalar (volumetric concentration of air bubble)
ϕ_0	Initial concentration at impingement section
ψ_L, ψ_R	MUSCL extrapolation advection flow variable
Q	Flow rate
r_d	Radius of air bubble
Re	Reynolds number
ρ	Water density
ρ_g	Density of air
ρ_l	Density of liquid
r_m	Data series with zero mean in digital filter
R_{uu}	Autocorrelation function for an array of data

S_i	Force per unit volume density
s_i	Source terms in Navier-Stokes equations
σ	Surface tension
$\tilde{S}_{ij}$	Strain rate tensors
$\left\|\tilde{S}_{ij}\right\|$	$\sqrt{2\tilde{S}_{ij}\tilde{S}_{ij}}$
$\tilde{\tilde{S}}_{ij}$	Test filtered strain rate tensors
s_ϕ	Source term in scalar transport equation
SS_{ti}	Source terms for bubbles in scalar transport equation
t	Time
τ_{ij}	Shear stress terms
τ_w	Wall shear stress
t_h	Jet thickness
θ	Crank Nicholson constant (*0, 0.5* or *1*)
T_{ij}	Test filtered shear stress terms
t^+	Dimensionless time in wall units
u, v, w	Velocities in *x*, *y*, and *z* direction of the Cartesian coordinate
$u*$	Approximate velocity
u', v', w'	Fluctuating velocity components
$\tilde{u}$	LES grid filtered velocities
$\bar{u}, \bar{\bar{u}}$	RANS velocities
$\tilde{\tilde{u}}$	LES test filtered velocities
Ue	Average velocity at upstream channel brink
u_i	Velocities vectors in tensor notation
u_{jet}	Average impinging velocity from the overfall
$u_\perp$	Velocity perpendicular to cell face
u_τ	Shear velocity

u_ξ, u_η, u_ζ	Velocities in generalized coordinate system
V_g	Velocity of gaseous phase
V_∞	Terminal velocity
V_{jet}	Jet impinging velocity
v_{jet}	Mean jet velocity
$\forall_g$	Volume of gaseous phase in a cell
$\forall_l$	Volume of liquid in a cell
V_r	Relative velocity
w_d	z-component of the Cartesian velocity, (*dz/dt*)
W_t	Weight of a single air bubble
x, y, z	Cartesian coordinate variables
X, Z	Coordinate position for overfall
x_i	Coordinate in tensor notation
x^+, y^+, z^+	Distance from wall in viscous wall units
ξ, η, ζ	Generalized coordinate variables
ξ_i, η_i, ζ_i	Grid metric terms from transformation

ABBREVIATIONS

CFD	Computational fluid dynamics
CFL	Courant-Friedrichs-Lewy stability condition
DNS	Direct numerical simulation
LES	Large eddy simulation
LHS	Left hand side of an equation
MAC	Marker-and-Cell method
PDE	Partial differential equation
PPE	Pressure Poisson equation
RANS	Reynolds Averaged Navier-Stokes equations
RHS	Right hand side of an equation
SGS	Subgrid scale
TDMA	Tridiagonal matrix algorithm or Thomas algorithm
VOF	Volume of fluid method
N-S	Navier-Stokes equations

CHAPTER 1

INTRODUCTION

1.1 - *Motivation*

The Navier-Stokes equations are thought to be the governing equations for all fluid motion and can be used to describe either laminar or turbulent flows [55]. Though simple to write, there is no formal analytical solution to these sets of equations. For that reason numerical modeling became an important part in the study of turbulent flows – to obtain an approximate solution.

Fluid dynamics can be described and categorized as laminar, transitional or turbulent flows. Laminar flows are the least complicated type of flow. The analytical solution of laminar flows can be obtained with ease and is widely available in most fluid mechanics textbooks ([55], [102], [111]). Transitional and turbulent flows are difficult to solve due to their stochastic and random behavior. Only approximate solutions are available for these two types of flow. For centuries, solving turbulent flow problems has been a challenge to engineers, mathematicians and physicists. It is no wonder why there is a million dollar prize, offered by The Clay Mathematics Institute of Cambridge (CMI), for obtaining analytical solution to the governing equation of turbulent and transitional flows. Unfortunately, most natural and engineering fluid flow problems fall into this category.

In civil engineering, three dimensional fluid dynamic problems are mostly academia based studies and their commercial uses are limited due to the scale of the problems. The application of numerical models to solve for practical fluids problems in civil engineering has still yet to catch up with its mechanical and chemical engineering counterparts. Although limited, the use of numerical modeling in civil engineering industrial application is not unheard of. The dam break analysis is one such example [128]. The needs, be it financial feasibility, convenience, or simply the interest in solving more complex problems, will be the catalyst to push the use of numerical modeling in civil engineering. The modeling of pump stations has seen such needs, especially in vortex

studies [36]. Figure 1-1 and Figure 1-2 shows sketches of a simple pump station with a planar plunging jet inflow.

In the past, studies of the pump stations relied on using physical model in laboratories. Scaled model had to be constructed and tested for results to determine if any problem would arise in the design. In scenarios where problems were identified, design geometry change requires the physical model be reconstructed and reexamined to find a workable configuration. This process demands tremendous effort and time. Furthermore the results obtained from physical model were extrapolated using similarities relationship to the prototype. The scaling is problematic due to some unaccounted non-dimensional parameters that are physically inconsistent between scale models and prototypes, and hence extrapolation of the laboratory data is not always reliable. In addition to modeling the hydrodynamics of pump stations, one of the more challenging problems is the air entrainment issue. Air bubbles appear as a result of impact, shear and mixing of two different fluid phases. In pump stations, air entrainment from an overfall is introduced, potentially affecting the operation of pumps. There is a need for a simple and dependable way of pump station modeling. Computer aided study seems to be a potentially advantageous alternative for this application and does not have the problems associated with physical modeling as mentioned prior.

The goal of this dissertation is to provide some insight on numerical air bubble modeling and at the same time offer an alternative to pump station modeling needed by the industry. This work is also motivated by the fact that numerical simulation of the air bubble and plunging jet problem provides a solution with much greater level of detail, if needed. A review of the literature reveals that the influence of air entrainment, forming air bubbles in the flow field, is still an area for active research. In addition, a numerical code which consists of three dimensional free surface capabilities along with the air bubble dynamics modeling is not available for plunging jet flows.

This research will focus on solving the three dimensional, free surface, plunging jet flows using the Large Eddy simulation, a dynamic subgrid-scale model, as well as modeling the transport of air bubbles. Air bubbles transport will be modeled via the transport equation, with some minor modification to account for buoyancy, with the effects added to the hydrodynamic model.

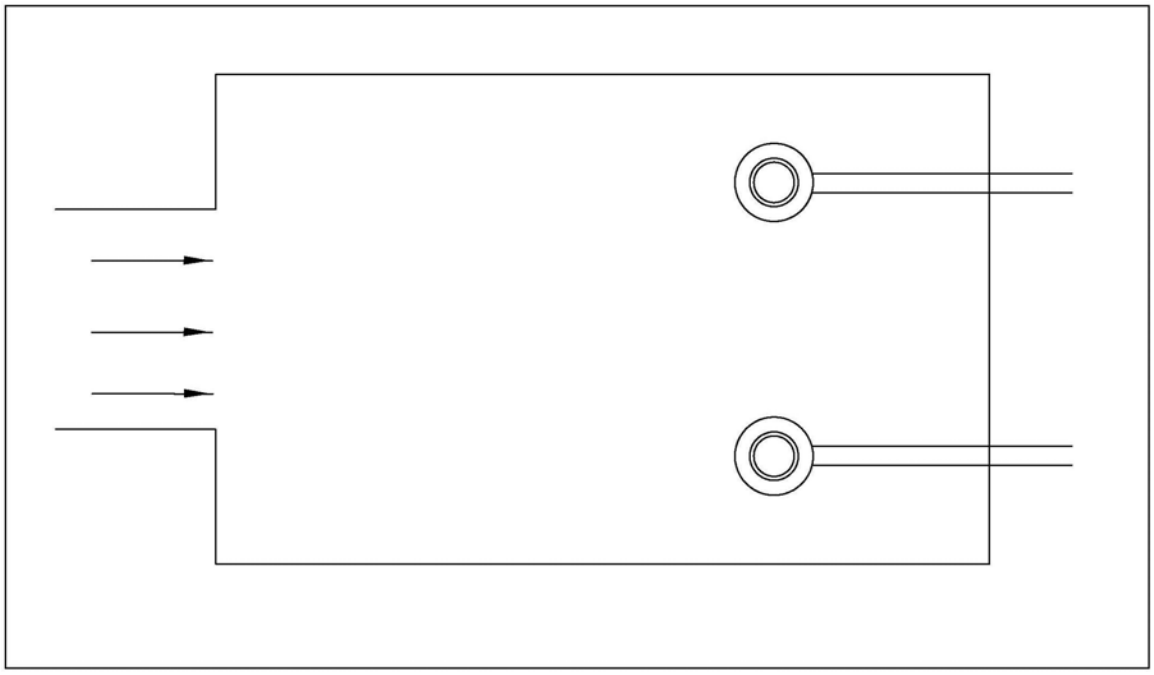

Figure 1-1: Plan view of a simple pump station.

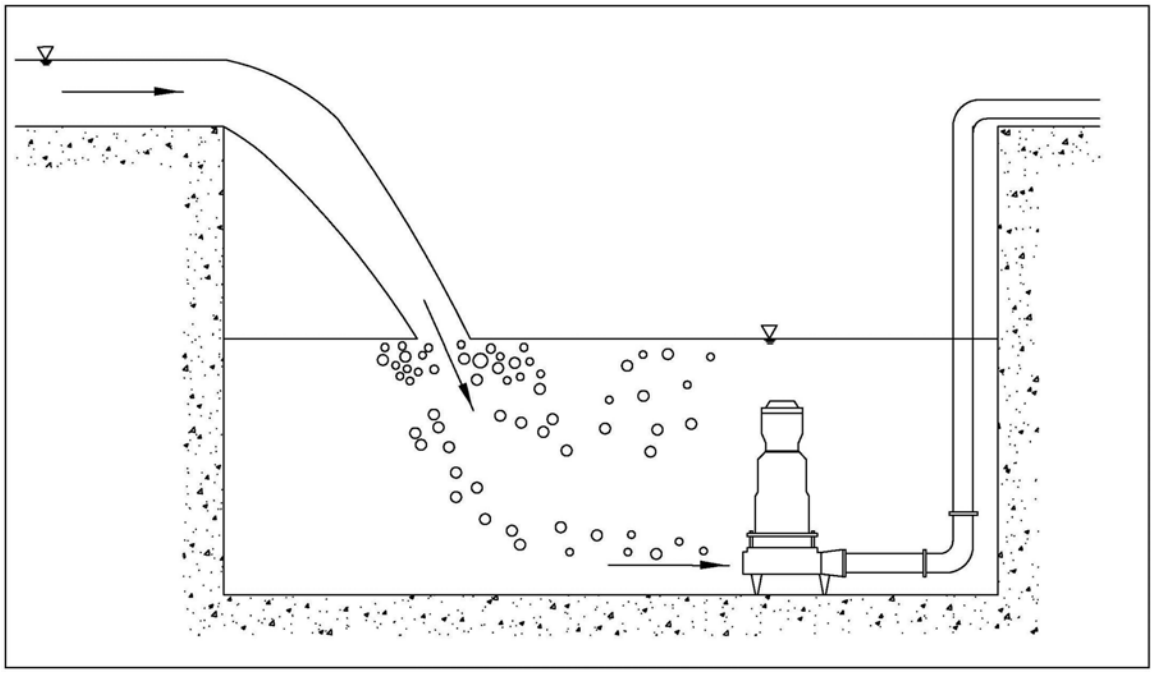

Figure 1-2: Side view of a simple pump station with plunging jet and air entrainment.

1.2 - *Literature Review*

Fluid dynamic problems can be solved via analytical, experimental or numerical methods. For more difficult problems, especially problems involving turbulent flows, analytical solutions are limited. Analytic solutions are often difficult to obtain due to the fact that the governing equations for most fluid flows are the Navier-Stokes equations and these are strictly non-linear [83]. While experimental work can be carried out to characterize and observe any type of flows, the difficulties and cost of setting up good experiments, as well as the tedious process of data collection make the experimental approach less desirable. The next option is to numerically obtain approximate solutions to the Navier-Stokes equations. The recent advancement of computational fluid dynamics (CFD) shows that the trend to model and observe complex fluid flows has shifted to the numerical approach.

Computational fluid dynamics is not a new field of study. In fact, the use of CFD dates as far back as the work by Thom [114] in 1933, to simulate the flow past a cylinder at low speeds. At present, CFD is considered a vital tool for the study of fluid flows, and is used to model the complex relationship of momentum, mass and energy transfer. Engineering problems that require computer modeling are typically *3*-dimensional turbulent flow problems where numerical solutions are difficult to obtain [59]. Large Eddy Simulation with dynamic subgrid scale models is the state of the art technology for fluid dynamics modeling and has matured to the extent that literature review in this area will yield hundreds, if not thousands, of publications. For that reason, a survey on this subject is deemed redundant. This work is merely adopting the existing numerical schemes and turbulence models for the present application, but integrating air bubble dynamics.

Numerical studies on air bubbles dynamics can be divided into two different classifications, namely the microscopic and macroscopic models [66]. The former category of bubble dynamic models involves modeling the detailed interaction of the air bubbles and water, and greatly emphasizes the interface tracking of the different fluids. The later classification models the average property of the two phases and treats the two fluids as a mixture. Examples of the microscopic modeling techniques are the interface

tracking type procedures such as the level set method [28], and the volume of fluid method [109]. Simulation of bubble dynamics in this category requires that the solution of the trajectories and the interface of the different fluid. This approach however is not suitable for macroscopic modeling of air bubbles, especially in large quantities, or in a larger geometry setting. The macroscopic modeling of air bubbles takes on another approach. The macroscopic modeling of air bubbles involves the assumption of mixture when dealing with the different phases. This leads to a simplified model which is computational inexpensive compared to the previously mentioned approach, but does not provide detailed information of individual air bubbles. Examples of this class of technique include the typical multiphase flow type model [30] and simplified multiphase types model [130]. This work will follow the macroscopic approach because detailed information of individual air bubbles is unnecessary and in addition, macroscopic models are better for large problems, as suggested by Deen *et al.* [44]

In the last decade, the simulation of bubble dynamics has received considerable interest from chemical engineers [30]. As some authors have suggested ([8], [49], [63]) more studies are needed in this area to better understand the different phase interaction and turbulence. The publication by Chen *et al.*, [30] in 2005, explicitly stated that most previous numerical studies compared their predictions of bubble dynamics qualitatively with experimental observations, and only few quantitative comparisons were made. Further search attempts to obtain data for numerical comparison of bubble simulation especially for multiphase flow type models, yielded little results, concurring to the claim by Chen *et al.* [30].

A vital part in dealing with air bubble dynamics is the determination of the terminal velocities for air bubbles of different sizes. Terminal velocities of air bubbles of different sizes had been reported by numerous researchers ([13], [25], [28], [62], [91], [122]). Mendelson [91] studied the terminal velocities of different sizes air bubbles in water and provided a relationship based on the wave theory. The relationship given in Mendelson's work was then compared with the experimental study by Haberman and Morton [62], and decent agreement was observed. However, Mendelson's equation found difficulties in correlating the terminal velocities of bubbles between *0.7 mm* and *3 mm* in radius. More recent work presented by Bozzano and Dante [13] provided the terminal velocity as a set

of equations. Bozzano and Dante provided the equations for the terminal velocity that also includes the effects of the drag coefficient of air bubbles and the deformation of air bubble during the course of its rise. The publication by Bozzano and Dante is significant for this work, as the equations were later adopted as part of the modeling of air bubble dynamics. The terminal velocities yielded from the equations corresponds well with chart given in Wallis's [122] book. Bozzano and Dente claimed to have also validated the equations using experimental data collected by Haberman and Morton [62].

Later, Chen and Fan [28] attempted simulation of the rise of a single air bubble numerically using the level set method. The detailed nature of Chen and Fan's work can be used as a good assessment for Bozzano and Dante's equations to further determine its validity. Numerical test of Bozzano and Dante's equations with *8 mm* diameter air bubbles was conducted, similar to Chen and Fan's paper. Result of the bubble position with time did not match exactly. Numerical results of Chen and Fan obtained a lower terminal velocity of approximately *18.75 cm/s*. Chen and Fan further commented that results obtained experimentally yielded a terminal velocity of *21 - 25 cm/s*, which agrees with the terminal velocity obtained from the theoretical equations presented by Bozzano and Dente. Comparisons were also made with the experimental data by Chanson [25] and satisfactory agreements were also obtained.

Air entrainment from different types of jets had been investigated experimentally by several different authors and research groups. Much of the work is experimentally based [8]. This is because the theoretical mechanism of air entrainment is not yet fully understood. In 1973, Van de Sande and Smith [119] published their hypothesis and experimental work on surface entrainment of air by high velocity water jets. In the publication, the authors claimed that a theory to predict the amount of air entrainment by a plunging jet was conceived for Weber numbers greater than *10* and the length based Reynolds numbers to be less than 5×10^5. According to the authors, the experiments conducted agreed well with the theory. The paper does relate the amount of air entrainment, but the jet was considered a round jet, constructed with cut pipe as the nozzle head. For the purpose of this work, interest was placed in plunging jet entrainment from a falling nappe, which has similar properties of a planar plunging jet. This difference prevented the use of the relationship in this work because the amount of air

entrainment not only depends on the impact velocity, but also geometry of the jet, angle of the jet, and *etc.*

In early studies of air bubble behavior, the difficulties faced by experimentalist were significant. The advancement of air bubble behavior studies was challenged by the task of keeping the size of bubbles uniform while experiments were being conducted [67]. Published work on air entrainment were scattered in different fields of studies. In 1993, Bin attempted a massive reorganization to compile most available experimental and theoretical studies on gas entrainment. The compilation resulted in a review article in chemical engineering science [8]. The review article was one of the more thorough surveys in this field and is a great reference to understand the prior work of other authors and research groups. In Bin's review paper, several major issues were discussed. The article highlighted the critical conditions for air entrainment, empirical relationships for quantification of air entrainment, characteristics of bubble dispersion, and practical applications of the plunging jets. Bin was well aware of the fact that the information presented in the review paper was insufficient to completely describe the complicated mechanics of air entrainment. In the conclusion of his paper, suggestions were made for more theoretical and experimental studies.

Shortly after the review paper by Bin, Hadjerioua *et al.* [63] attempted to relate quantity of air entrainment by falling nappes via a *2*-dimensional semi-analytic analysis. In the article, the momentum equations for the air boundary layer were analytically analyzed through the use of integral method to determine the thickness of air layer being dragged down the plunge pool. The analytic results were then compared with experiment data done by Ervine and Elsawy [50]. Hadjerioua *et al.* [63] realized the discrepancy between the experimental data and the results of the analytical analysis was due to the nappe thickness not being factored into consideration, and thus used the least squares regression analysis to help establish a general relationship. This semi-analytic equation showed good agreement to experimental results. However the use of this equation is limited to nappes less than *60 mm* thick, and thus not a practical range for an overfall nappe in pumping stations. Since Hadjerioua *et al.* [63] obtained the semi-analytical relationship based on Ervine and Elsawy's [50] experimental data, further investigations to find work derived from the same data sets yielded Mason's publication [86] which

provided a well documented equation used by other researchers ([18], [90]) to quantify the amount of air entrainment. According to the literature [18], the equation found in Mason's publication is a typical formula used for air bubble quantification of air in plunge pool scour research, which resembles the characteristics of the plunging jet for a free overfall drop in this study. Therefore, that equation was chosen for air entrainment quantification for this research.

Hubert Chanson, an experimentalist from University of Queensland, Brisbane, Australia has numerous publications ([16], [21], [22], [26]), regarding air entrainment quantity and mechanism. In his book [24], Chanson summarized some empirical equations that other experimentalist had published in a tabular form with their respective application limits. Chanson then proceeds to collapse the equations down into three acceptable forms according to the range of impact velocity of the jet. The equation provided by Mason [86] was not listed in Chanson's summary, however it agreed with the general form of equations suggested in Chanson's analysis. Chanson also mentioned that the equations were meant for deep receiving pools with no or slow motion. Other publications by Chanson could be used to enhance the understanding of the mechanism of air entrainment, mainly on location of air bubble generation and quantification under extreme cases of an overfall jet. Other forms of equations provided by Cumming and Chanson ([38] , [39]) in their papers regarding jet entrainment requires a priori knowledge of the flow field and are rather complex.

In a more recent article published by Melo [90], a section was dedicated for historical review of the air entrainment quantity at the free jet impact section. In the review, the work of Chanson was not referenced. It is believed that Melo's work was independent from Chanson's. Therefore, it is not surprising that Melo suggested an entirely different set of equations for the air entrainment quantities. The equations in the article, as cited from Ervine and Falvey [51], Ervine *et al.* [52], Bohrer and Abt [10] and Bohrer *et al.* [11], suggest that the equations were also empirical relations deduced from experimental work. Comparison were made using the equations provided in Melo's work with that of Mason's [86] mentioned earlier, and significant differences in the results were observed. The differences may have been attributed to incomplete understanding of the applicable

limits of the equations in Melo's paper. In addition, the equations provided by Melo's might not have been appropriate for an overfall plunging jet.

In 2002, El Hammoumi *et al*. [49] quoted the conclusion from Bin's work that there was still no successful theoretical approach to predict the critical entrainment velocity from a vertical plunging jet. That work provided valuable experimental studies, but little theoretical work. They claimed that there existed no concrete theoretical insight of modeling the plunging jet air entrainment. Currently researchers rely only on empirical relations. Due to the lack of understanding of the air entrainment mechanism, numerical simulation of the plunging jet into receiving pool of fluid is scarce.

Early numerical work in civil engineering to simulate large scale water-bubble flow began with the work of Zarrati [130] in 1994. Zarrati simulated a channel flow with an upstream sluice gate, and air bubbles were artificially injected into the channel. In his simulation, Zarratti assumed a mixture fluid, where air and water were treated as a single continuum. Zarrati included the buoyancy effect of the air bubbles by adjusting the Schmidt number, a parameter in his turbulent model. The problem of doing such adjustments is that physics of the buoyant effect is buried in the turbulence modeling part of the numerical simulation, and is somewhat questionable. Based on force balance between the air bubbles and water, the buoyancy effects should be included as part of the resolvable scale. Despite that, the numerical simulation yielded satisfactory results. Zarrati's work helped pave the way for other civil engineering researchers to advance the knowledge of simulating multiphase flows and in some sense suggested the possibility of treating the two different fluids as a single mixture continuum.

A group of researchers in China, Xu *et al.* [124], attempted the simulation of a *3*-dimensional turbulent aerated flow in a plunge pool. The two-phase or multiphase flow equations were used. For turbulent modeling, the *k*-epsilon (*k*-ε) model was employed, similar to Zarrati's publication. Xu *et al.* [124] claimed that the results of the simulation for air concentration are reasonable and the trend was predicted correctly. The numerical results were compared to the experimental work done by Dong Zhiyong for his Ph.D dissertation, written in Chinese. In another paper published by Xu *et al.*, [125], a *3*-dimensional simulation of plunge pool was attempted, however, with the exclusion of air

entrainment. According to Xu *et al.* [125] and Liao *et al.* [81], there was no other prior numerical simulation of a *3*-dimensional plunge pool prior to theirs.

Formal numerical simulation for a *3*-dimensional pump intake structure was first published in 1998 by *Constantinescu et al.*[36]. Their emphasis established the formation of free-surface and wall-attached vortices. The turbulent model used was the k-ε model. Follow up work by Ansar *et al.* [3], Constantinescu and Patel [37], Rajendran *et al*. [100], Constantinescu and Patel [36], Tokyay and Constantinescu [116] and Tokyay and Constantinescu [117] focused on the formation of vortices and not air entrainment and transport. In addition, no plunging jet was simulated, which differs from this investigation.

As there is still no known formal investigation on the work to numerically simulate an inflow of a plunging jet with air bubbles dynamics, this area of study is worth investigating.

1.3 - *Organization of Dissertation*

This dissertation is organized into several different chapters that provide the required background, development, validation tests, results as well as conclusion of this research. The dissertation begins (Chapter 1) with an introduction to the current trend in fluid dynamics modeling, the necessity for this work to be conducted and the objective of this dissertation. A literature survey was carried out to provide some insights of previous work that is relevant to this dissertation.

Chapter 2 is devoted to the hydrodynamics and turbulence subgrid scale modeling. This chapter outlines the theoretical development and methodology of the hydrodynamics model. The choice of the governing equations and the turbulence models used in this work were presented, alongside the discussions of the procedure in arriving at their usable forms. The numerical schemes for the spatial and temporal discretization, and boundary conditions were also detailed in the second chapter of this dissertation.

Following that, Chapter 3 describes the dynamics and quantification of air bubbles. Since there is no formal theoretical quantification of air bubbles from a plunging jet, the known empirical relationship from literature was provided. More importantly, Chapter 3

attempts to simplify and link the air bubble dynamics with the governing equations for the hydrodynamics, as well as provide a simplified model for the tracking of air bubble concentration using the scalar transport equation. The modifications to the scalar transport equation to accommodate the movement of air bubbles and the relationship of the bubble dynamics to the momentum equations were shown. Limitations of the bubble transport model were discussed.

Following the discussion of the air bubbles, the next chapter (Chapter 4) provides information on the overfall, namely the size, location, angle and the impinging velocities of the jet. The information consists of theoretical and empirical equations from literature and was used as part of the essential boundary conditions for the modeling. This short chapter also describes the implementation of these boundary conditions at the impact section of the free overfall jet.

In pursuing this work, considerable amount of effort was put into testing the numerical code. Verification and validation of the code was done systematically and is shown in Chapter 5. Results from the numerical simulation were compared with analytic, numerical and available experimental data from the literature. At the closing of this chapter, a simple case of a plunging jet flow was simulated.

Finally, Chapter 6 concludes the dissertation by providing a summary of the achievements and recommendations for future research. The expected end product of this research consists of a numerical model which is robust for simulation of either a turbulent or laminar conditions for the plunging jet flows, with free surface and air bubbles transport capability.

CHAPTER 2

BASIC THEORY AND GOVERNING EQUATIONS

Chapter 2 provides some historical background on the development of the continuity and Navier-Stokes equations, leading to the turbulent Large Eddy Simulation decomposed equations used in this work. In this chapter, the governing equations, along with the numerical schemes, were discussed in depth. Additionally, the spatial and temporal discretization techniques for the different terms in the governing equations were also detailed.

A review of the turbulent closure models for the LES equations is provided to enhance the reader's understanding on turbulent modeling. This includes some formal explanation of eddy viscosity type subgrid scale models, namely the classical Smagorinsky closure model and the dynamic Smagorinsky closure model, emphasizing more on the dynamic Smagorinsky closure model. The implementation of some general boundary conditions will be discussed.

2.1 - *Governing Equations*

Mass balance is one of the most fundamental concepts in solving engineering problems. The idea of mass balance is widely used to describe mass transfer from one location to another in a given control section or volume. The continuity equation uses the concept of mass balance to describe the rate of mass transfer in a continuous body of fluid in a control volume. In a *3*-dimensional Cartesian coordinate system, the continuity equation is written as

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \tag{2.1.1}$$

where, *u, v* and *w* are velocities in the *x*, *y*, and *z* direction of the Cartesian coordinate. The continuity equation, also known as the divergence-free condition or incompressibility constraint, will be one of the governing equations in this work. The

others governing equations consists of the Navier-Stokes momentum balance equations. The momentum balance equations were named after a French scientist, C.L.M.H. Navier and English Physicist Sir George Stokes [55]. The Navier-Stokes equations are invaluable tool to analyze fluid motion problems, as it is the foundation for most computational fluid dynamics codes. Navier-Stokes equations can be thought of as a set of second order partial differential equations (PDE) that are derived based on physics of fluid motion, namely the conservation of momentum. In Cartesian coordinate system, the complete set of Navier-Stokes equations for incompressible fluid consists of the three momentum balance in the three different directions, *x*, *y* and *z* coordinates respectively.

x-direction momentum balance:

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)=-\frac{\partial P}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+f_x \tag{2.1.2}$$

y-direction momentum balance:

$$\rho\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}\right)=-\frac{\partial P}{\partial y}+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right)+f_y \tag{2.1.3}$$

z-direction momentum balance:

$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}\right)=-\frac{\partial P}{\partial z}+\mu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)+f_z \tag{2.1.4}$$

where

ρ is the density

μ is the viscosity, assumed constant,

P is the pressure

f_x, f_y, f_z are the Coriolis parameter in the three respective directions

Most *3*-dimensional fluid dynamics problem can be described using the Navier-Stokes equations as the governing equation. There are basically three major philosophies in approximating the solution to the Navier-Stokes equations. They are the direct numerical simulation, time averaged equations, and ensemble averaged equations. The

basis of the different approaches and its advantages and disadvantages will be explained in the subsequent sections.

This work will focus on the ensemble averaged type Navier-Stokes equations or the Large Eddy Simulation equations. The Reynolds Averaged Navier-Stokes (RANS) equations are time averaged equations while the Large Eddy Simulation (LES) set of equations are the ensemble averaged family of the Navier-Stokes equations. The LES equations can be written as follows

$$\frac{\partial \tilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\tilde{u}_i \tilde{u}_j + \tau_{ij}\right) = -\frac{1}{\rho}\frac{\partial \tilde{P}}{\partial x_i} + \nu_T \frac{\partial}{\partial x_j}\left[\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i}\right] \tag{2.1.5}$$

For simplicity, the tensor notation is used to represent the three spatial directions, where substituting *i=1,2,3* and *j=1,2,3* with summation over repeated indices will produce the full set of *3*-dimensional Navier-Stokes momentum balance equations in a spatially filtered field. Equation (2.1.5) is not much different from equations (2.1.2)-(2.1.4) in terms of solving for the variables, however, the former contains physics and statistics which can be used to avoid having to perform a full direct numerical simulation (but requiring a turbulent closure model).

The spatially filtered Navier-Stokes equations will be coupled with the incompressibility constraint or continuity equation, shown as equation (2.1.1), to describe the dynamics in the flow field. For cases of laminar flow with known boundary conditions, problems can be solved using the two sets of equations. However, for a turbulent flow problem, a closure for the equation is required. This closure will be further discussed in sections 2.3 and 2.4. For the scope of this research, only incompressible fluid is considered.

2.2 – *Background on Turbulence Modeling*

Ideally numerical approximation of the Navier-Stokes equations can be obtained by discretizing the equation into very fine grids and directly solving the discretized equations. That is exactly the concept of the direct numerical simulation (DNS). Direct numerical simulation is by far the most accurate, however most naïve way to simulate

flows of any type. DNS, when implemented appropriately, does not require modeling of any terms and is therefore arguably the most complete method to solve the N-S equations. In order to capture the large scales and small scales of the slightest changes in velocities in a flow field, the resolution of the spatial and temporal grids has to be extremely high.

Using DNS to perform calculations of a flow field for a finite amount of time requires tremendous amount of arithmetic operations. Sometimes, the amount of arithmetic required, is in the order of the Reynolds number cube $\mathcal{O}(Re^3)$ [88], is so overwhelming that even the current state of the art computers are insufficient to resolve a practical engineering problem. Literatures have shown that simple problems can be carried out using DNS [56]; nonetheless the use of DNS is limited due to the affordability of computation power. Researchers therefore resort to the knowledge of statistics to simplify the Navier-Stokes equations to avoid performing a DNS. Among the well known statistical based Navier-Stokes equations are the Reynolds Averaged Navier-Stokes (RANS) equations and the Large Eddy Simulation (LES) equations, each having its own advantages and disadvantages.

The RANS is a temporal averaged variation of the Navier-Stokes equations, proposed as an alternative to performing a full DNS. The idea of using statistics to tackle the turbulent flow problem had been documented in the mid 1890's [88]. RANS was developed based on statistical analysis of turbulent flows. The governing equation of the RANS is very similar to that of the DNS, but the physical significance of each term is not. In the RANS formulation of the N-S equations, the governing equation was time averaged. As a result, another set of equations were obtained to describe the time averaged properties of the fluid dynamics, while the fluctuating components of the turbulence is modeled.

The use of a turbulent model to describe the fluctuating quantities greatly reduces the computational effort because coarser grids are now possible. The arithmetic requirement of RANS is much less than the DNS, with at most, the equivalent order to the Reynolds number, $\mathcal{O}(Re)$ [88]. Because time averaging is used in the RANS equations, large and small scales of turbulence are modeled at the same time. This implies that the solution obtained from RANS may not be accurate due to the extensive modeling [88]. In contrast,

the LES formulation separate the turbulence scales and tend to resolve the large scale turbulence and model the small scale turbulence. The LES has found favor because it resolves the important scale in turbulence which contains important physics of flow and models the less important scales, *i.e.* the small scales. Therefore the LES approach is chosen for this study. Formal derivation of the LES set of equations will be provided in the subsequent section (section 2.3).

2.3 - *Large Eddy Simulation (LES)*

Large eddy simulation was also developed based on statistical averages of turbulent flows. It is no surprise that the governing equations for LES are strikingly similar to that of the RANS equations. Unlike the RANS, the LES resolves the large scale and models the small scale quantities. This can be done by taking the ensemble average of the N-S equations, by passing the N-S equations through a spatial filter with a known filter size. Hence the turbulent scales larger than the filter size are resolvable while turbulent scales smaller than the filter size are not.

The fundamental LES decomposition is

$$u(x,t) = \tilde{u}(x,t) + u'(x,t) \tag{2.3.1}$$

In this decomposition, $\tilde{u}(x,t)$ are the resolved scale or large scale quantity and the term $u'(x,t)$ are known as the unresolved scale, or small scale quantities. The resolved and unresolved scale quantities are both spatial and temporal dependant. The properties that the RANS approach utilized to simplify its derivation are different from the LES decomposition, or more specifically

$$\text{RANS:} \quad \bar{\bar{u}} = \bar{u} \quad \text{and} \quad \bar{u}' = 0$$

$$\text{LES:} \quad \tilde{\tilde{u}} \neq \tilde{u} \quad \text{and} \quad \tilde{u}' \neq 0$$

In these relationships, the overbar denoted time averaging while tilde represents ensemble averaging. Due to this difference, the derivation of LES becomes more complicated than the RANS. The fundamental idea of the LES was based on scale separation statistics. As discussed in section 2.2, the RANS models all scales of turbulence, LES resolves the

large scale while modeling the small scales. This is the main advantage of LES formulation over the RANS formulation, as less modeling introduces less modeling errors.

Before a formal derivation of the LES equations, it is helpful to first introduce the filtering concept. In the LES procedure, the main interest is to resolve the larger scales motion of the flow. The scales smaller than the wavelength of the spatial step, Δx_i needs to be extracted from the governing equation. To accomplish this, a low-pass filter is used. The generalized form of the filter [80] is

$$\overline{fi}(x,t) = \int fi(y,t)\overline{G}(x-y)dy = \int fi(x-y,t)\overline{G}(y)dy \tag{2.3.2}$$

where $\overline{G}$ is defined as the function for the low-pass filter and fi is the vector or scalar quantity of interest. The filter function can take on various forms, for example, the Gaussian filter, box filter, cutoff filter, *etc.* [53].

The derivation of the LES sets of equations involves passing the Navier-Stokes equations through the previously defined filter, now called the grid filter. For simplicity, the equations will again be written in tensor notation. The filtered equation of continuity is obtained by using equation (2.3.1), namely

$$\frac{\partial u_j}{\partial x_j} = \frac{\partial(\tilde{u}_j + u_j')}{\partial x_j} \tag{2.3.3}$$

Since the divergence-free condition has to be satisfied on large and small scales, therefore, by definition,

$$\frac{\partial u_j'}{\partial x_j} = 0 \tag{2.3.4}$$

and hence the filtered continuity equation becomes

$$\frac{\partial \tilde{u}_j}{\partial x_j} = 0 \tag{2.3.5}$$

Formulation of the momentum equation is not as straightforward. This is because the cross terms are non-linear. Neglecting the Coriolis effect, the grid filtered momentum equation becomes

$$\frac{\partial \tilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\widetilde{u_i u_j}\right) = -\frac{1}{\rho}\frac{\partial \tilde{p}}{\partial x_i} + \nu\frac{\partial}{\partial x_j}\left[\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i}\right] \tag{2.3.6}$$

The first term on the left hand side of equation (2.3.6) can be left as is, due to the commutative property of the filter with temporal derivatives. The treatment of the right hand side is similar to that of the continuity equation. The difficulties lie in the non-linear terms where such direct simplification is not possible. These non-linear terms are

$$\frac{\partial}{\partial x_j}\left(\widetilde{u_i u_j}\right) = \frac{\partial}{\partial x_j}\left[\widetilde{\left(\tilde{u}_i + u_i'\right)\left(\tilde{u}_j + u_j'\right)}\right] \tag{2.3.7}$$

and it becomes apparent that because $\tilde{u}' \neq 0$, equation (2.3.7) may only be expanded to

$$\frac{\partial}{\partial x_j}\left[\widetilde{\left(\tilde{u}_i + u_i'\right)\left(\tilde{u}_j + u_j'\right)}\right] = \frac{\partial}{\partial x_j}\left[\widetilde{\tilde{u}_i\tilde{u}_j + \tilde{u}_i u_j' + \tilde{u}_j u_i' + u_i' u_j'}\right] \tag{2.3.8}$$

and none of the terms in equation (2.3.8) can be eliminated to simplify the expression to a usable form

$$\frac{\partial}{\partial x_j}\left(\tilde{u}_i\tilde{u}_j\right) \tag{2.3.9}$$

Therefore, the nonlinear term have to be treated differently. The only choice to formulate equation (2.3.7) into equation (2.3.9) is to introduce an additional modeled term τ_{ij}, such that

$$\tilde{u}_i\tilde{u}_j + \left[\widetilde{\left(\tilde{u}_i + u_i'\right)\left(\tilde{u}_j + u_j'\right)} - \tilde{u}_i\tilde{u}_j\right] \equiv \tilde{u}_i\tilde{u}_j + \tau_{ij} \tag{2.3.10}$$

τ_{ij} will now become the LES subgrid scale terms and will be modeled using a subgrid scale (SGS) closure model for the LES formulation. Substitute the above relationship back into the filtered Navier-Stokes equations, equation (2.3.6) gives

$$\frac{\partial\tilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\tilde{u}_i\tilde{u}_j + \left[\widetilde{\left(\tilde{u}_i + u_i'\right)\left(\tilde{u}_j + u_j'\right)} - \tilde{u}_i\tilde{u}_j\right]\right) = -\frac{1}{\rho}\frac{\partial\tilde{p}}{\partial x_i} + \nu\frac{\partial}{\partial x_j}\left[\frac{\partial\tilde{u}_i}{\partial x_j} + \frac{\partial\tilde{u}_j}{\partial x_i}\right] \tag{2.3.11}$$

and the equation will then leads to the usable form of the ensemble averaged Navier-Stokes equations

$$\frac{\partial\tilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\tilde{u}_i\tilde{u}_j + \tau_{ij}\right) = -\frac{1}{\rho}\frac{\partial\tilde{p}}{\partial x_i} + \nu\frac{\partial}{\partial x_j}\left[\frac{\partial\tilde{u}_i}{\partial x_j} + \frac{\partial\tilde{u}_j}{\partial x_i}\right] \tag{2.3.12}$$

With the exception of the subgrid scale terms, all other variable are resolvable and thus can be calculated directly. The following section, section 2.4, details the closure model for the LES set of equations.

Note that the arithmetic requirement for the LES is not as much as the DNS, however is greater than that of RANS. The approximate arithmetic of the LES is to the order of $\mathcal{O}(Re^2)$ [88]. One important feature of the LES approach is that as the grid size approaches zero ($\Delta x_i \to 0$), the LES set of equations will converge to the DNS equations. The only questionable part of the LES approximation is the small scale turbulent modeling, which is still a topic of research. Until a better solution is found to resolve all scales of turbulence, LES and its variants will remain a solid approximation for the Navier-Stokes equations.

2.4 - *Subgrid Scale Model*

Turbulent flow modeling involves approximation of the Navier-Stokes equations as well as providing a good closure model to mathematically describe the modeled velocities fluctuations. The complexity of turbulent flow simulation results from the way the Navier-Stokes equations are being solved [64], as well as the construction of the closure model. Failure to provide a closure model will result in an ill-posed mathematical problem, with the one exception of performing a DNS. As mentioned earlier, DNS is still unaffordable and not practical for solving modern engineering problems. Thus, the aim of a subgrid scale (SGS) model is to mimic the turbulence fluctuation statistics and convey the information back to the large resolved scale.

Boussinesq paved the way for most turbulent models with his hypothesis that turbulent stresses are linearly proportional to mean strain rate [88]. The Boussinesq eddy viscosity assumption utilizes the property that the turbulent stresses are proportional to the mean velocity gradients, similar to the viscous stresses in laminar flows. Unlike the fluid viscosity, the turbulent eddy viscosity is not a fluid property. The turbulent eddy viscosity arises as simplified terms which relate the fluctuating velocities to the Navier-Stokes equations. Boussinesq's eddy viscosity assumption reduces the complexity of calculating the turbulent stresses based on the fluctuating velocities and instead relates the stresses as eddy viscosities, as the modeling of the eddy viscosity are far simpler and can be conveniently be correlated back to fundamental statistics. Many subgrid scale

models used in LES adopted this hypothesis and were built from this idea. The classical Smagorinsky models, dynamic models and mixed models were among the better known SGS models that took advantage of this hypothesis. The mathematical formulation of the classical Smagorinsky model is given as

$$\tau_{ij} = -2\nu_t \tilde{S}_{ij} \tag{2.4.1}$$

where,

$$\nu_t = C_s{}^2 \Delta^2 \left|\tilde{S}_{ij}\right| \tag{2.4.2}$$

$$\tilde{S}_{ij} = \frac{1}{2}\left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} \right) \tag{2.4.3}$$

$$\left|\tilde{S}_{ij}\right| = \sqrt{2\tilde{S}_{ij}\tilde{S}_{ij}} \tag{2.4.4}$$

In the classical Smagorinsky subgrid scale model, C_s, called the Smagorinsky model coefficient, is a parameter multiplier that typically ranges in between *0.06* to *0.2,* and is problem dependant ([47], [53], [57]).

Though the classical Smagorinsky subgrid scale model is simple, a clear disadvantage to this model is that the Smagorinsky model coefficient is neither a constant, nor a parameter. Many questions can be raised regarding the problem dependant nature of C_s. Therefore, prescribing the value of C_s as a constant coefficient might not be the best way to compute the spatially dependant stress terms. In addition, it makes little sense to assign the same C_s value for the whole flow field. Velocity fluctuations are different in the various regions of the domain, and hence so should C_s.

Nevertheless, numerous computational work ([5], [7], [69], [85]) has been successfully done using specified values of C_s. Specifying a single value for any simulation reduces the computation effort, but the simulation results can only be a crude approximation. This deficiency observed from the classical Smagorinsky SGS formulation makes it less desirable for the current work. However, because of its simplicity, the classical Smagorinsky model will still be considered as a secondary turbulent model.

The dynamic subgrid scale model attempts to address the shortcoming of having a constant coefficient. Instead of predefining the parameter C_s, the strategy of the dynamic

model is to locally evaluate C_s using information from neighboring cells. To achieve this, Germano *et al*. [57] was one of the first to apply a coarser filter to the grid filtered equation of motion and established a mathematical relationship for the variables at the two different filter levels. Similar to the procedure in deriving the LES set of equations, the LES set of equations (Equation (2.3.12)) is again passed through another filter, a larger filter called the test filter, resulting in similar set of equations

$$\frac{\partial \tilde{\tilde{u}}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\tilde{\tilde{u}}_i \tilde{\tilde{u}}_j + T_{ij}\right) = -\frac{1}{\rho}\frac{\partial \tilde{\tilde{p}}}{\partial x_i} + \nu \frac{\partial}{\partial x_j}\left[\frac{\partial \tilde{\tilde{u}}_i}{\partial x_j} + \frac{\partial \tilde{\tilde{u}}_j}{\partial x_i}\right] \tag{2.4.5}$$

Previously the definition of the shear stresses was, $\tau_{ij} \equiv \widetilde{u_i u_j} - \tilde{u}_i \tilde{u}_j$. Similarly, for the test filter $T_{ij} \equiv \widetilde{\widetilde{u_i u_j}} - \tilde{\tilde{u}}_i \tilde{\tilde{u}}_j$. Applying the Germano identity [57], the relationship of τ_{ij} and T_{ij} can be established

$$L_{ij} = T_{ij} - \tilde{\tau}_{ij} = \widetilde{\tilde{u}_i \tilde{u}_j} - \tilde{\tilde{u}}_i \tilde{\tilde{u}}_j \tag{2.4.6}$$

L_{ij} is better known as the Leonard stresses. L_{ij} can also be quantified using the resolved scale velocities. The terms T_{ij} and τ_{ij} contains the spatially varying C_s coefficients.

Recall that for incompressible fluid,

$$\tau_{ij} = -2\Delta^2 C_s^{\,2} \left|\tilde{S}_{ij}\right| \tilde{S}_{ij} \tag{2.4.7}$$

and in the same way, T_{ij} can be expressed using the twice filtered quantities

$$T_{ij} = -2\tilde{\Delta}^2 C_s^{\,2} \left|\tilde{\tilde{S}}_{ij}\right| \tilde{\tilde{S}}_{ij} \tag{2.4.8}$$

Since the Leonard stress terms, L_{ij}, is obtained from $\tilde{\tau}_{ij}$, τ_{ij} will have to be averaged based on the same test filter scale. Performing the ensemble averaging gives

$$\tilde{\tau}_{ij} = -2\Delta^2 \widetilde{C_s^{\,2} \left|\tilde{S}_{ij}\right| \tilde{S}_{ij}} \tag{2.4.9}$$

It is important to point out that the ensemble averaging performed on τ_{ij} is not the same as simply passing the grid filtered variables through the test filter. The first term on the right hand side of equation (2.4.6) should result from the same ensemble average. Substituting $\tilde{\tau}_{ij}$ and T_{ij} into equation (2.4.6), with $L_{ij} = T_{ij} - \tilde{\tau}_{ij}$, gives

$$L_{ij} = -2\tilde{\Delta}^2 C_s^{\,2} \left|\tilde{\tilde{S}}_{ij}\right| \tilde{\tilde{S}}_{ij} - \left(-2\Delta^2 \widetilde{C_s^{\,2} \left|\tilde{S}_{ij}\right| \tilde{S}_{ij}}\right) \tag{2.4.10}$$

A standard assumption is that C_s will be scale invariant, *i.e.* C_s will remain "approximately the same" within the test filer width, hence

$$C_s^2(\Delta) \approx C_s^2(\tilde{\Delta}) \approx C_s^2 \tag{2.4.11}$$

and thus,

$$-2\Delta^2 \widetilde{C_s^2 \left|\tilde{S}_{ij}\right| \tilde{S}_{ij}} = -2\Delta^2 C_s^2 \widetilde{\left|\tilde{S}_{ij}\right| \tilde{S}_{ij}} \tag{2.4.12}$$

Further manipulation can be made to factor out C_s

$$L_{ij} = 2C_s^2 \left[-\tilde{\Delta}^2 \left|\tilde{\tilde{S}}_{ij}\right| \tilde{\tilde{S}}_{ij} + \Delta^2 \widetilde{\left|\tilde{S}_{ij}\right| \tilde{S}_{ij}} \right] \tag{2.4.13}$$

For simplicity, the expression is trimmed down to

$$L_{ij} = 2CM_{ij} \tag{2.4.14}$$

where, $M_{ij} = -\tilde{\Delta}^2 \left|\tilde{\tilde{S}}_{ij}\right| \tilde{\tilde{S}}_{ij} + \Delta^2 \widetilde{\left|\tilde{S}_{ij}\right| \tilde{S}_{ij}}$ and $C = C_s^2$. The variation of C within the test filter can now be minimized by using the least squares approximation. The procedure [82] will be shown for completeness, and is as follows:

Let e be the least squared error term within the test filter

$$e = \left(L_{ij} - 2CM_{ij}\right)^2 \tag{2.4.15}$$

$$e = L_{ij}^2 - 4CL_{ij}M_{ij} + 4C^2 M_{ij}^2 \tag{2.4.16}$$

Minimizing the squared error with respect to C leads to

$$\frac{\partial e}{\partial C} = \frac{\partial}{\partial C}\left(L_{ij}L_{ij} - 4CL_{ij}M_{ij} + 4C^2 M_{ij}M_{ij}\right) = 0 \tag{2.4.17}$$

$$-4L_{ij}M_{ij} + 8CM_{ij}M_{ij} = 0 \tag{2.4.18}$$

$$C = \frac{L_{ij}M_{ij}}{2M_{ij}M_{ij}} \tag{2.4.19}$$

The algebraic tensor operations for equation (2.4.19) restrict cancellation of terms in the numerator and denominator. Instead, the numerator and denominator should be calculated separately before the division operation is carried out. The steps shown above will complete the dynamic subgrid scale model, thus providing a closure for the LES set of equations based on flow statistics, and not a specified value.

In this work, the test filter variables are computed using the trapezoidal rule as used in Ding [46] and Zang *et al.* [129]. The implementation of the test filter follows three succession steps ([46], [129])

$$\begin{aligned}
\lambda^{*}_{j,k,l} &= \frac{1}{4}\left(\lambda_{j+1,k,l} + 2\lambda_{j,k,l} + \lambda_{j-1,k,l}\right) \\
\lambda^{**}_{j,k,l} &= \frac{1}{4}\left(\lambda^{*}_{j,k+1,l} + 2\lambda^{*}_{j,k,l} + \lambda^{*}_{j,k-1,l}\right) \\
\tilde{\tilde{\lambda}}_{j,k,l} &= \frac{1}{4}\left(\lambda^{**}_{j,k,l+1} + 2\lambda^{**}_{j,k,l} + \lambda^{**}_{j,k,l-1}\right)
\end{aligned} \tag{2.4.20}$$

where λ represent the grid filtered variables.

The dynamic SGS model is much more appealing than the traditional Smagorinsky SGS model because of its capability to calculate C_s based on local spatial information instead of a user specified constant as in the case of the classical Smagorinsky model. Hence, local physics is represented in the calculations. Another distinct feature of the dynamic subgrid scale model is that the magnitude of the dynamic coefficient diminishes accordingly as the wall is approached. The traditional Smagorinsky subgrid scale model failed to produce this characteristic, and consequently requires the use of a damping function. The dynamic subgrid scale model is an improvement over the traditional Smagorinsky subgrid scale model and for this reason and is used as the primary turbulent model in this work.

2.5 - *Anisotropic Grids Treatment for Smagorinsky Model*

Variants of Smagorinsky type SGS models use an equivalent length scale as its filter length, shown as Δ in equation (2.4.2). In a three dimensional models, a common practice is to follow Deardorff [43], which computes the equivalent length scale as

$$\Delta = \left(\Delta_x \Delta_y \Delta_z\right)^{\frac{1}{3}} \tag{2.5.1}$$

In cases where the filter length is exaggerated in certain a direction, it is problematic to define the correct length scale which the SGS filter operates [104]. In such instance, the SGS filter is defined by more than one length scale. Calculating Δ based on the cubic

root of the grid sizes will be problematic because the equivalent length scale will not be a good representation of the grid discretization in the direction of the smallest grid size and thus create more uncertainties and errors to the SGS model. Scotti and Meneveau [103] suggested an adjustment factor for the Smagorinsky type models as a remedy. The function $f(ag_1,ag_2)$ is used as a multiplication factor to the Smagorinsky length scale, and can be approximated by

$$f(ag_1, ag_2) = \cosh\sqrt{\frac{4}{27}\left[(\ln ag_1)^2 - \ln ag_1 \ln ag_2 + (\ln ag_2)^2\right]} \tag{2.5.2}$$

where, $ag_1 = \frac{\Delta_1}{\Delta_2}$, $ag_2 = \frac{\Delta_2}{\Delta_3}$

The general rule for ag_1 and ag_2 is that the grid length scale is always ordered as $\Delta_1 \leq \Delta_2 \leq \Delta_3$. The adjustment factor, $f(ag_1,ag_2)$ will be multiplied with the equivalent length scale, equation (2.5.1), to give an "adjusted" length scale. For clarity, when Equation (2.5.2) is coupled with equation (2.4.1) the resulting expression is

$$\tau_{ij} = -2[C_s\Delta(f(ag_1, ag_2))]^2 \left|\widetilde{S}_{ij}\right| \widetilde{S}_{ij} \tag{2.5.3}$$

In this work, equation (2.5.2) is applied to the traditional Smagorinsky model, the secondary turbulent model.

2.6 - *Finite Volume Numerical Scheme*

To perform numerical calculations of the Navier-Stokes equations and the continuity equation, the equations have to be discretized. The finite volume discretization is used for this study. The finite volume discretization is a localized mass and momentum balance preserving numerical scheme which is a desirable trait for the application of this research. A locally conservative scheme is important because for open channel flow problems, the free surface location varies with time. This variation often requires the vertical grid size to be adjusted based on local flux balance. Local conservations will be advantageous because mass and momentum balance is necessary in every cell to achieve good accuracy in critical parts of the flow field.

The complete filtered Navier-Stokes equations can be rewritten in the most general form, using tensor notation, as

$$\frac{\partial \tilde{u}_i}{\partial t}+\frac{\partial(\tilde{u}_i\tilde{u}_j)}{\partial x_j}=-\frac{1}{\rho}\frac{\partial \tilde{p}}{\partial x_i}+\frac{\mu}{\rho}\frac{\partial}{\partial x_j}\left(\frac{\partial \tilde{u}_i}{\partial x_j}+\frac{\partial \tilde{u}_j}{\partial x_i}\right)+\frac{\partial \tilde{\tau}_{ij}}{\partial x_j}-\frac{\partial(g\tilde{h})}{\partial x_i}-\frac{g}{\rho_o}\int_z^h\frac{\partial \rho}{\partial x_i}dz+f\tilde{u}_j \quad (2.6.1)$$

where,

τ_{ij} are the shear stress terms divided by fluid density

g is the gravitational force

h is the depth of water

ρ_o is the reference density, density of the fluid

f is the Coriolis parameter, and

p is the dynamic pressure term

The pressure tensor term shown in equation is the dynamic pressure, or the pressure deviation from hydrostatic pressure. The Navier-Stokes equations can be rewritten as follows, analogous to Bradford and Katopodes [14].

$$\frac{\partial \tilde{\boldsymbol{u}}}{\partial t}+\frac{\partial \tilde{\mathbf{f}}}{\partial x}+\frac{\partial \tilde{\mathbf{g}}}{\partial y}+\frac{\partial \tilde{\mathbf{h}}}{\partial z}=s \quad (2.6.2)$$

where $\tilde{\boldsymbol{u}}$ represent the different velocities vectors in the Cartesian coordinate system. The other variables shown in equation (2.6.2) are shown as follows

$$\mathbf{f}=\begin{pmatrix} u^2-\tau_{xx}+p+gh \\ uv-\tau_{xy} \\ uw-\tau_{xz} \end{pmatrix} \qquad \mathbf{g}=\begin{pmatrix} uv-\tau_{xy} \\ v^2-\tau_{yy}+p+gh \\ vw-\tau_{yz} \end{pmatrix}$$

$$\mathbf{h}=\begin{pmatrix} uw-\tau_{xz} \\ vw-\tau_{yz} \\ w^2-\tau_{zz}+p \end{pmatrix} \qquad s=\begin{pmatrix} -\frac{g}{\rho_0}\int_z^h\frac{\partial \rho}{\partial x}dz^*+fv \\ -\frac{g}{\rho_0}\int_z^h\frac{\partial \rho}{\partial y}dz^*-fu \\ 0 \\ \end{pmatrix}$$

The pressure terms, p, found in the $\mathbf{f}$, $\mathbf{g}$ and $\mathbf{h}$ are the dynamic pressures divided by the density, ρ_o.

Here the shear stresses are defined as

$$\tau_{xx} = 2\nu_T \frac{\partial u}{\partial x},\ \tau_{yy} = 2\nu_T \frac{\partial v}{\partial y},\ \tau_{zz} = 2\nu_T \frac{\partial w}{\partial z}$$

$$\tau_{xy} = \nu_T \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right),\ \tau_{xz} = \nu_T \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right),\ \tau_{yz} = \nu_T \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right)$$

Unlike Bradford and Katopodes [14], the vertical and horizontal eddy diffusivities are not treated separately in the diffusion terms. Hence, the assumption of isotropic turbulence must hold. The other implicit assumptions are Boussinesq approximation and Boussinesq eddy viscosity assumption. The Boussinesq approximation is used when there are differences in density in the fluid, namely due to salinity or temperature changes in the domain. This assumption will take into consideration the densities gradients and adjust for the buoyancy forces.

A transformation can be performed to the equations to convert from the Cartesian coordinate system to a generalized coordinate system for computation. The purpose of the transformation is to create flexibility of the code to take on different types of geometry and bathymetry for the governing equations. The transformation into general coordinates also provide convenience in calculating odd shaped cells and defines each cell as a unit cube, and additionally, placing the velocity vectors perpendicular to its respective cell faces, as shown in Figure 2-1.

Due to the numerical scheme and coordinate transformation, scalar variables such as the dynamic pressure, temperature or concentrations are located at the center of the cell.

The transformation is done once before the numerical computation with the transform information stored for later use at each time step. Once computations are done, the general coordinate system is transformed back to the Cartesian coordinate system as the post processing procedure. For the vertical direction, the transformation variables are recalculated at every time step after each update of the free surface.

The resulting governing equations in generalized coordinate are:

$$\frac{Du_\xi}{Dt} + F_\xi + H_\xi + P_\xi = s_\xi \tag{2.6.3}$$

$$\frac{Du_\eta}{Dt} + F_\eta + H_\eta + P_\eta = s_\eta \tag{2.6.4}$$

$$\frac{Du_\zeta}{Dt} + F_\zeta + P_\zeta = s_\zeta \tag{2.6.5}$$

where, F_i are the convection and diffusion terms,

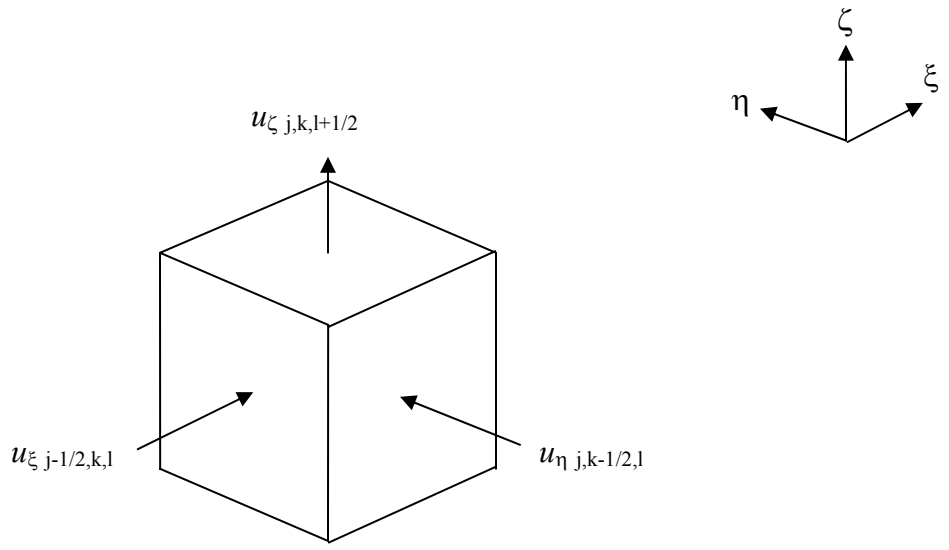

Figure 2-1: A typical computational cell in the generalized coordinate system.

H_i are the free surface gradient terms,

P_i are the pressure gradient terms, and

s_i are the source or sink terms

The fluxes and shear stresses are lumped into the variable F_i in equations (2.6.3) through (2.6.5) and are defined as

$$F_\xi = \frac{\partial}{\partial \xi}\left(u_\xi{}^2 - \tau_{\xi\xi}\right) + \frac{\partial}{\partial \eta}\left(u_\xi u_\eta - \tau_{\xi\eta}\right) + \frac{\partial}{\partial \zeta}\left(u_\xi u_\zeta - \tau_{\xi\zeta}\right) - \frac{dz}{dt} w_d \frac{\partial u_\xi}{\partial \zeta}$$

$$F_\eta = \frac{\partial}{\partial \xi}\left(u_\xi u_\eta - \tau_{\xi\eta}\right) + \frac{\partial}{\partial \eta}\left(u_\eta{}^2 - \tau_{\eta\eta}\right) + \frac{\partial}{\partial \zeta}\left(u_\eta u_\zeta - \tau_{\eta\zeta}\right) - \frac{dz}{dt} w_d \frac{\partial u_\eta}{\partial \zeta}$$

$$F_\zeta = \frac{\partial}{\partial \xi}\left(u_\xi u_\zeta - \tau_{\xi\zeta}\right) + \frac{\partial}{\partial \eta}\left(u_\eta u_\zeta - \tau_{\eta\zeta}\right) + \frac{\partial}{\partial \zeta}\left(u_\zeta{}^2 - \tau_{\zeta\zeta}\right) - \frac{dz}{dt} w_d \frac{\partial u_\zeta}{\partial \zeta}$$

The variable w_d is an additional term arisen from the deformation of the vertical grid due to the free surface change, and will be shown later in section 2.8, equation (2.8.3). The free surface terms are

$$H_\xi = g\left(l_{\xi\xi}\frac{\partial h}{\partial \xi} + l_{\xi\eta}\frac{\partial h}{\partial \eta}\right)$$

$$H_\eta = g\left(l_{\xi\eta}\frac{\partial h}{\partial \xi} + l_{\eta\eta}\frac{\partial h}{\partial \eta}\right)$$

The pressure terms are

$$P_\xi = l_{\xi\xi}\frac{\partial p}{\partial \xi} + l_{\xi\eta}\frac{\partial p}{\partial \eta} + l_{\xi\zeta}\frac{\partial p}{\partial \zeta}$$

$$P_\eta = l_{\xi\eta}\frac{\partial p}{\partial \xi} + l_{\eta\eta}\frac{\partial p}{\partial \eta} + l_{\eta\zeta}\frac{\partial p}{\partial \zeta}$$

$$P_\zeta = l_{\xi\zeta}\frac{\partial p}{\partial \xi} + l_{\eta\zeta}\frac{\partial p}{\partial \eta} + \left(l_{\zeta\zeta} + \zeta_z\right)\frac{\partial p}{\partial \zeta}$$

All shear stress terms can be expanded into the form

$$\tau_{\xi\xi} = 2\nu_T\left(l_{\xi\xi}\frac{\partial u_\xi}{\partial \xi} + l_{\xi\eta}\frac{\partial u_\xi}{\partial \eta} + l_{\xi\zeta}\frac{\partial u_\xi}{\partial \zeta}\right)$$

$$\tau_{\eta\eta} = 2\nu_T\left(l_{\xi\eta}\frac{\partial u_\eta}{\partial \xi} + l_{\eta\eta}\frac{\partial u_\eta}{\partial \eta} + l_{\eta\zeta}\frac{\partial u_\eta}{\partial \zeta}\right)$$

$$\tau_{\zeta\zeta} = 2\nu_T \left(l_{\xi\zeta} \frac{\partial u_\zeta}{\partial \xi} + l_{\eta\zeta} \frac{\partial u_\zeta}{\partial \eta} + l_{\zeta\zeta} \frac{\partial u_\zeta}{\partial \zeta} + \zeta_z{}^2 \frac{\partial u_\zeta}{\partial \zeta} \right)$$

$$\tau_{\xi\eta} = \nu_T \left(\frac{\partial}{\partial \xi}\left(u_\xi l_{\xi\eta} + u_\eta l_{\xi\xi} \right) + \frac{\partial}{\partial \eta}\left(u_\xi l_{\eta\eta} + u_\eta l_{\xi\eta} \right) + \frac{\partial}{\partial \zeta}\left(u_\xi l_{\eta\zeta} + u_\eta l_{\xi\zeta} \right) \right)$$

$$\tau_{\xi\zeta} = \nu_T \left(\frac{\partial}{\partial \xi}\left(u_\xi l_{\xi\zeta} + u_\zeta l_{\xi\xi} \right) + \frac{\partial}{\partial \eta}\left(u_\xi l_{\eta\zeta} + u_\zeta l_{\xi\zeta} \right) + \frac{\partial}{\partial \zeta}\left(u_\xi l_{\zeta\zeta} + u_\zeta l_{\xi\zeta} \right) + \zeta_z{}^2 \frac{\partial u_\xi}{\partial \zeta} \right)$$

$$\tau_{\eta\zeta} = \nu_T \left(\frac{\partial}{\partial \xi}\left(u_\eta l_{\xi\zeta} + u_\zeta l_{\xi\eta} \right) + \frac{\partial}{\partial \eta}\left(u_\eta l_{\eta\zeta} + u_\zeta l_{\eta\eta} \right) + \frac{\partial}{\partial \zeta}\left(u_\eta l_{\zeta\zeta} + u_\zeta l_{\eta\zeta} \right) + \zeta_z{}^2 \frac{\partial u_\eta}{\partial \zeta} \right)$$

where,

$$l_{\xi\xi} = \xi_x{}^2 + \xi_y{}^2 \qquad l_{\xi\eta} = \xi_x \eta_x + \xi_y \eta_y \qquad l_{\xi\zeta} = \xi_x \zeta_x + \xi_y \zeta_y$$

$$l_{\eta\eta} = \eta_x{}^2 + \eta_y{}^2 \qquad l_{\eta\zeta} = \eta_x \zeta_x + \eta_y \zeta_y \qquad l_{\zeta\zeta} = \zeta_x{}^2 + \zeta_y{}^2$$

Relationships of the contravariant velocities, u_ξ , u_η , and u_ζ , to the Cartesian velocities are

$$u_\xi = u\xi_x + v\xi_y$$

$$u_\eta = u\eta_x + v\eta_y$$

$$u_\zeta = u\zeta_x + v\zeta_y + w\zeta_z$$

The vertical Cartesian velocity components are not shown for u_ξ and u_η. This is due to the orthogonality of the horizontal coordinates with the vertical coordinate where the grid metric terms ξ_z and η_z vanishes. The source terms follows the same transformation and are

$$s_\xi = s_x \xi_x + s_y \xi_y$$

$$s_\eta = s_x \eta_x + s_y \eta_y$$

$$s_\zeta = s_x \zeta_x + s_y \zeta_y + s_z \zeta_z$$

The transformation assumes a linear mapping from the Cartesian coordinate space (x, y, z) to the generalized coordinate space (ξ, η, ζ), resulting in the grid metric terms ξ_x, ξ_y , η_x, η_y, ζ_x, ζ_y, and ζ_z appearing in the equations of the generalized coordinate.

The continuity equation retains the same form, but with different coordinate variables.

$$\frac{\partial u_\xi}{\partial \xi} + \frac{\partial u_\eta}{\partial \eta} + \frac{\partial u_\zeta}{\partial \zeta} = 0 \tag{2.6.6}$$

The free surface equation (see section 2.8) after coordinate transformation, is

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial \xi}\left(\int_{-d}^{h} u_\xi dz\right) + \frac{\partial}{\partial \eta}\left(\int_{-d}^{h} u_\eta dz\right) = 0 \tag{2.6.7}$$

The resulting equations (equations (2.6.3) through (2.6.7)) can then be used for the hydrodynamic calculations.

There are two different ways to obtain the LES sets of equations in the generalized coordinate. The first way is to pass the equation through the grid filter before the coordinate transformation. The second way is to filter the Navier-Stokes equations after the coordinate transformation is performed. In this work, the former approach is taken. For convenience, the variables are assumed to be LES decomposed variables and the tilde notation will be omitted from this point onwards.

2.7 - *Time Splitting Scheme*

Time discretization is as important as the spatial discretization. There are three basic types of time discretization, fully explicit, semi implicit and fully implicit. In this study, the Crank Nicholson (semi-implicit) scheme is used to discretize the pressure and free surface terms. The Crank Nicholson time discretization is a second order accurate method, thus preserving the overall discretization to second order accuracy.

The viscous and advection terms were discretized using the explicit predictor-corrector method. This approach is stable, and does not require additional smoothing filters ([9], [14]). As a result of the time discretization, the *3*-dimensional momentum equations becomes

$$
\begin{aligned}
&\frac{u_\xi^{n+1}-u_\xi^{n}}{\Delta t}+F_\xi^{n+\frac{1}{2}}+\theta\left(H_\xi+P_\xi\right)^{n+1}+\left(1-\theta\right)\left(H_\xi+P_\xi\right)^{n}=s_\xi^{n+\frac{1}{2}}\\
&\frac{u_\eta^{n+1}-u_\eta^{n}}{\Delta t}+F_\eta^{n+\frac{1}{2}}+\theta\left(H_\eta+P_\eta\right)^{n+1}+\left(1-\theta\right)\left(H_\eta+P_\eta\right)^{n}=s_\eta^{n+\frac{1}{2}}\\
&\frac{u_\zeta^{n+1}-u_\zeta^{n}}{\Delta t}+F_\zeta^{n+\frac{1}{2}}+\theta P_\zeta^{n+1}+\left(1-\theta\right)P_\zeta^{n}=s_\zeta^{n+\frac{1}{2}}
\end{aligned}
\qquad (2.7.1)
$$

where, the superscript *n*, *n+1/2* and *n+1* are current, intermediate and new time levels, respectively. The use of a predictor–corrector scheme, demands the computation of variables at the intermediate time level, *n+1/2*. The velocities at the intermediate time level can be calculated from the predictor step using the velocities at the current time level. In short, the equations are

$$
\begin{aligned}
u_\xi^{n+\frac{1}{2}}&=u_\xi^{n}-\frac{\Delta t}{2}\left(F_\xi+H_\xi+P_\xi-s_\xi\right)^{n}\\
u_\eta^{n+\frac{1}{2}}&=u_\eta^{n}-\frac{\Delta t}{2}\left(F_\eta+H_\eta+P_\eta-s_\eta\right)^{n}\\
u_\zeta^{n+\frac{1}{2}}&=u_\zeta^{n}-\frac{\Delta t}{2}\left(F_\zeta+P_\zeta-s_\zeta\right)^{n}
\end{aligned}
\qquad (2.7.2)
$$

Once these velocities are obtained, the advection, diffusion, and source terms at the intermediate time level can be estimated. These intermediate values of the variables will then be used to compute the final velocities. But, before the computation of the final velocities for each time step, the free surface and pressure terms have to be included as part of the iterative procedure.

To clearly show the vertical implicit treatment of the diffusion terms, equation (2.7.1) can be written as

$$
\begin{aligned}
&\frac{u_\xi^{n+1}-u_\xi^{n}}{\Delta t}+F_\xi^{n+\frac{1}{2}}-\theta\left(w_d\frac{\partial u_\xi}{\partial\zeta}+\frac{\partial m_\xi}{\partial\zeta}-H_\xi-P_\xi\right)^{n+1}-\left(1-\theta\right)\left(w_d\frac{\partial u_\xi}{\partial\zeta}+\frac{\partial m_\xi}{\partial\zeta}-H_\xi-P_\xi\right)^{n}=s_\xi^{n+1/2}\\
&\frac{u_\eta^{n+1}-u_\eta^{n}}{\Delta t}+F_\eta^{n+\frac{1}{2}}-\theta\left(w_d\frac{\partial u_\eta}{\partial\zeta}+\frac{\partial m_\eta}{\partial\zeta}-H_\eta-P_\eta\right)^{n+1}-\left(1-\theta\right)\left(w_d\frac{\partial u_\eta}{\partial\zeta}+\frac{\partial m_\eta}{\partial\zeta}-H_\eta-P_\eta\right)^{n}=s_\eta^{n+1/2}\\
&\frac{u_\zeta^{n+1}-u_\zeta^{n}}{\Delta t}+F_\zeta^{n+\frac{1}{2}}-\theta\left(w_d\frac{\partial u_\zeta}{\partial\zeta}+\frac{\partial m_\xi}{\partial\zeta}+\frac{\partial m_\eta}{\partial\zeta}+\frac{\partial m_\zeta}{\partial\zeta}-P_\zeta\right)^{n+1}-\\
&\left(1-\theta\right)\left(w_d\frac{\partial u_\zeta}{\partial\zeta}+\frac{\partial m_\xi}{\partial\zeta}+\frac{\partial m_\eta}{\partial\zeta}+\frac{\partial m_\zeta}{\partial\zeta}-P_\zeta\right)^{n}=s_\zeta^{n+1/2}
\end{aligned}
$$

where the vertical diffusion terms were separated from the horizontal diffusion terms. The vertical diffusion terms are

$$m_\xi = \nu_T \frac{\partial u_\xi}{\partial \zeta} \qquad m_\eta = \nu_T \frac{\partial u_\eta}{\partial \zeta} \qquad m_\zeta = 2\nu_T \frac{\partial u_\zeta}{\partial \zeta}$$

Since these vertical diffusion terms are treated implicitly, discretization of these terms will result in tridiagonal matrix systems. The solution to the tridiagonal matrix system can be obtained using the tridiagonal matrix algorithm (TDMA), also known as the Thomas algorithm.

Next, some explanations are provided on the free surface and dynamic pressure treatment, and how these equations can be integrated into the momentum balance equations.

2.8 - *Free Surface Treatment*

A free surface boundary is different from the other boundary conditions because the computation of the free surface is not as straightforward as prescribing known fluxes or constants at the boundary. A free surface can deform with time and further complicate the analysis. The Marker-and-Cell (MAC) method [33], volume of fluid (VOF) method [76], level set method [105], and *etc.* are among the available techniques to solve for free surface deformation. In this work, the free surface was computed using a deforming grid system in the vertical direction to account for the adjustment of depth caused the change in free surface. The deforming grid in the vertical direction required additional terms to be added to the Navier-Stokes momentum equations to account for the change of free surface elevation in time. The temporal dependant terms for grid movement are written as

$$\frac{Du}{Dt} = \frac{\partial u}{\partial t} + \frac{\partial u}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial u}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial u}{\partial z}\frac{\partial z}{\partial t} \tag{2.8.1}$$

Since the horizontal grids does not vary in time, the second and third terms at the right hand side of the equation above goes to zero, hence

$$\frac{Du}{Dt} = \frac{\partial u}{\partial t} + \frac{\partial u}{\partial z}\frac{\partial z}{\partial t} \tag{2.8.2}$$

In rare cases where the horizontal grids also changes with time, all terms will have to be included in the analysis. The left over terms will adjust for the change in the vertical grids. Rearranging the equation,

$$\frac{\partial u}{\partial t} = \frac{Du}{Dt} - \frac{\partial u}{\partial z}\frac{\partial z}{\partial t} = \frac{Du}{Dt} - w_d \frac{\partial u}{\partial z} \tag{2.8.3}$$

The upper case differential operator represents the Lagrangian frame of reference while the lower case represents the Eulerian frame of reference. Substituting the equation into the Navier-Stokes equations, one will obtain the Navier-Stokes momentum balance equations with the deforming grid adjustment. The free surface variables can now be factored into the computation by using the principle of mass balance. The basic continuity equation has the form

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.8.4}$$

The derivation on the free surface equation requires the use of integration of the continuity equation. Performing the integration to the continuity equation in the vertical coordinate gives

$$\int_{-d}^{h} \frac{\partial u}{\partial x} dz + \int_{-d}^{h} \frac{\partial v}{\partial y} dz + \int_{-d}^{h} \frac{\partial w}{\partial z} dz = 0 \tag{2.8.5}$$

From the Leibnitz rule, the equation becomes

$$\frac{\partial}{\partial x}\int_{-d}^{h} u dz + \frac{\partial}{\partial y}\int_{-d}^{h} v dz + \frac{\partial}{\partial z}\int_{-d}^{h} w dz = 0 \tag{2.8.6}$$

Further simplifying the equation gives

$$\frac{\partial}{\partial x}\int_{-d}^{h} u dz + \frac{\partial}{\partial y}\int_{-d}^{h} v dz + \left[w\right]_{-d}^{h} = 0 \tag{2.8.7}$$

The last term in equation (2.8.7) is defined by $\left[w\right]_{-d}^{h} = w(h) - w(-d)$ and utilizing the impermeable bottom condition, the equation can be rewritten to

$$\frac{\partial}{\partial x}\int_{-d}^{h} u dz + \frac{\partial}{\partial y}\int_{-d}^{h} v dz + w(h) = 0 \tag{2.8.8}$$

The kinematic free surface boundary condition states that, $w(h)=\dfrac{\partial h}{\partial t}$. Thus, using the kinematic free surface boundary condition, one obtains the equation used to determine the free surface location

$$\frac{\partial h}{\partial t}+\frac{\partial}{\partial x}\left(\int_{-d}^{h} u dz\right)+\frac{\partial}{\partial y}\left(\int_{-d}^{h} v dz\right)=0 \tag{2.8.9}$$

Equation (2.8.9) can now be transformed in the general coordinate system. The transformation yields

$$\frac{\partial h}{\partial t}+\frac{\partial}{\partial \xi}\left(\int_{-d}^{h} u_{\xi} dz\right)+\frac{\partial}{\partial \eta}\left(\int_{-d}^{h} u_{\eta} dz\right)=0 \tag{2.8.10}$$

Bradford and Katopodes [14] and Bradford and Sanders [15] have used this approach with success, without resorting to artificial diffusion to stabilize the free surface computation. The limitation to this approach is that this procedure is unable to cope with wave breaking phenomena. If the surface does break, other techniques should be used to approximate the free surface.

The temporal discretization of the free surface equation using the θ method resulted in

$$\begin{aligned}&\frac{h^{n+1}-h^{n}}{\Delta t}+\theta\left(\frac{\partial}{\partial \xi}\left(\int_{-d}^{h} u_{\xi} dz\right)+\frac{\partial}{\partial \eta}\left(\int_{-d}^{h} u_{\eta} dz\right)\right)^{n+1}\\&+(1-\theta)\left(\frac{\partial}{\partial \xi}\left(\int_{-d}^{h} u_{\xi} dz\right)+\frac{\partial}{\partial y}\left(\int_{-d}^{h} u_{\eta} dz\right)\right)^{n}=0\end{aligned} \tag{2.8.11}$$

Spatially discretizing equation (2.8.11) then yields,

$$\begin{aligned}&h_{j,k}{}^{n+1}+\theta\Delta t\sum_{l=1}^{nl}\begin{bmatrix}\left(\tilde{u}_{\xi}{}^{n+1}\Delta z\right)_{j+1/2,k,l}-\left(\tilde{u}_{\xi}{}^{n+1}\Delta z\right)_{j-1/2,k,l}\\+\left(\tilde{u}_{\eta}{}^{n+1}\Delta z\right)_{j,k+1/2,l}-\left(\tilde{u}_{\eta}{}^{n+1}\Delta z\right)_{j,k-1/2,l}\end{bmatrix}=\\&h_{j,k}{}^{n}-(1-\theta)\Delta t\sum_{l=1}^{nl}\begin{bmatrix}\left(u_{\xi}{}^{n}\Delta z\right)_{j+1/2,k,l}-\left(u_{\xi}{}^{n}\Delta z\right)_{j-1/2,k,l}\\+\left(u_{\eta}{}^{n}\Delta z\right)_{j,k+1/2,l}-\left(u_{\eta}{}^{n}\Delta z\right)_{j,k-1/2,l}\end{bmatrix}\end{aligned} \tag{2.8.12}$$

where, ξ, η and ζ are direction contiguous to the *j*, *k*, and *l* indices respectively, and will be used repeatedly from here on. At the *n+1* time level, the velocities required are approximate velocities that do not take into consideration the dynamic pressure terms, *i.e.* setting *p=0*. Since the dynamic pressure terms are not considered in the momentum

equation (see first 3 equations in time splitting scheme, equations (2.7.1)), only intermediate values of F and s are known, the approximate velocities were estimated from

$$\tilde{u}_{\xi}^{n+1} = u_{\xi}^{n} - \Delta t\left(F_{\xi} - s_{\xi}\right)^{n+1/2} - \left(1-\theta\right)\Delta t H_{\xi}^{n} - \theta\Delta t H_{\xi}^{n+1} \tag{2.8.13}$$

$$\tilde{u}_{\eta}^{n+1} = u_{\eta}^{n} - \Delta t\left(F_{\eta} - s_{\eta}\right)^{n+1/2} - \left(1-\theta\right)\Delta t H_{\eta}^{n} - \theta\Delta t H_{\eta}^{n+1} \tag{2.8.14}$$

$$\tilde{u}_{\zeta}^{n+1} = u_{\zeta}^{n} - \Delta t\left(F_{\zeta} - s_{\zeta}\right)^{n+1/2} \tag{2.8.15}$$

As mentioned earlier, because the horizontal velocities were calculated independent of the diffusion terms, discretizing the diffusion terms in each direction resulted in a simple tridiagonal matrix, A, for each direction. In matrix form, the horizontal components of the approximate velocities in the ξ direction is

$$AU_{\xi}^{n+1} + H_{\xi}^{n+1} = U_{\xi}^{n+1} \tag{2.8.16}$$

upon further mathematical manipulation, equation (2.8.16) gives

$$A^{-1}AU_{\xi}^{n+1} + A^{-1}H_{\xi}^{n+1} = A^{-1}U_{\xi}^{n+1} \tag{2.8.17}$$

Simplifying equation (2.8.17) yields

$$U_{\xi}^{n+1} = A^{-1}U_{\xi}^{n} - A^{-1}H_{\xi}^{n+1} \tag{2.8.18}$$

Analogous relationship for the U_{η} vector can be obtained. In the solution process, the inverse of the tridiagonal, matrix, A^{-1} is required. The inverse is calculated from the Thomas algorithm (TDMA). U^{n+1} is now a function of U^{n} and H^{n+1}. Substituting the approximate velocities back into the free surface equation, the equation becomes

$$
\begin{aligned}
&h_{j,k}{}^{n+1} - (\theta\Delta t)^2 \left\{ \begin{array}{l} \left(H_\xi{}^{n+1} D\right)_{j+1/2,k} - \left(H_\xi{}^{n+1} D\right)_{j-1/2,k} \\ + \left(H_\eta{}^{n+1} D\right)_{j,k+1/2} - \left(H_\eta{}^{n+1} D\right)_{j,k-1/2} \end{array} \right\} = \\
&h_{j,k}{}^{n} - (1-\theta)\Delta t \sum_{l=1}^{nl} \left[\begin{array}{l} \left(u_\xi{}^{n} \Delta z\right)_{j+1/2,k,l} - \left(u_\xi{}^{n} \Delta z\right)_{j-1/2,k,l} \\ + \left(u_\eta{}^{n} \Delta z\right)_{j,k+1/2,l} - \left(u_\eta{}^{n} \Delta z\right)_{j,k-1/2,l} \end{array} \right] - \\
&\theta\Delta t \sum_{l=1}^{nl} \left[\begin{array}{l} \left(\left(u_\xi{}^{n} - \Delta t \left(F_\xi - s_\xi \right)^{n+1/2} - (1-\theta)\Delta t H_\xi{}^{n} \right) \Delta z \right)_{j+1/2,k,l} - \\ \left(\left(u_\xi{}^{n} - \Delta t \left(F_\xi - s_\xi \right)^{n+1/2} - (1-\theta)\Delta t H_\xi{}^{n} \right) \Delta z \right)_{j-1/2,k,k} + \\ \left(\left(u_\eta{}^{n} - \Delta t \left(F_\eta - s_\eta \right)^{n+1/2} - (1-\theta)\Delta t H_\eta{}^{n} \right) \Delta z \right)_{j,k+1/2,l} - \\ \left(\left(u_\eta{}^{n} - \Delta t \left(F_\eta - s_\eta \right)^{n+1/2} - (1-\theta)\Delta t H_\eta{}^{n} \right) \Delta z \right)_{j,k-1/2,l} \end{array} \right]
\end{aligned}
\tag{2.8.19}
$$

The right hand side of equation (2.8.19) consists of the known variables and can be calculated. The left hand side of the equation is free surface values at the current time level. The variable *D* denotes total water depth at a cell face.

As shown in Bradford and Katopodes [14], discretizing the equation with a centered difference scheme (shown in Appendix A), yields an equation with nine unknown free surface terms, *h*. Equation (2.8.19) can then be summarized as

$$
\begin{aligned}
&a_1 h_{j-1,k-1}{}^{n+1} + a_2 h_{j-1,k}{}^{n+1} + a_3 h_{j-1,k+1}{}^{n+1} + a_4 h_{j,k-1}{}^{n+1} + a_5 h_{j,k}{}^{n+1} + a_6 h_{j,k+1}{}^{n+1} + a_7 h_{j+1,k-1}{}^{n+1} \\
&+ a_8 h_{j+1,k}{}^{n+1} + a_9 h_{j+1,k+1}{}^{n+1} = RHS
\end{aligned}
\tag{2.8.20}
$$

The coefficients a_1 through a_9 are as follows, with a_5 being the coefficient at the center of the stencil (diagonal term in the matrix)

$$
a_1 = -g(\theta\Delta t)^2 \frac{\left(l_{\xi\eta} D\right)_{j-1/2,k} + \left(l_{\xi\eta} D\right)_{j,k-1/2}}{4}
$$

$$
a_3 = -g(\theta\Delta t)^2 \frac{\left(l_{\xi\eta} D\right)_{j-1/2,k} + \left(l_{\xi\eta} D\right)_{j,k+1/2}}{4}
$$

$$
a_7 = -g(\theta\Delta t)^2 \frac{\left(l_{\xi\eta} D\right)_{j+1/2,k} + \left(l_{\xi\eta} D\right)_{j,k-1/2}}{4}
$$

$$
a_9 = -g(\theta\Delta t)^2 \frac{\left(l_{\xi\eta} D\right)_{j+1/2,k} + \left(l_{\xi\eta} D\right)_{j,k+1/2}}{4}
$$

$$a_2 = -g\left(\theta\Delta t\right)^2\left\{\left(l_{\xi\xi}D\right)_{j-1/2,k} + \frac{\left(l_{\xi\eta}D\right)_{j,k-1/2} - \left(l_{\xi\eta}D\right)_{j,k+1/2}}{4}\right\}$$

$$a_4 = -g\left(\theta\Delta t\right)^2\left\{\left(l_{\eta\eta}D\right)_{j,k-1/2} + \frac{\left(l_{\xi\eta}D\right)_{j-1/2,k} - \left(l_{\xi\eta}D\right)_{j+1/2,k}}{4}\right\}$$

$$a_6 = -g\left(\theta\Delta t\right)^2\left\{\left(l_{\eta\eta}D\right)_{j,k+1/2} + \frac{\left(l_{\xi\eta}D\right)_{j+1/2,k} - \left(l_{\xi\eta}D\right)_{j-1/2,k}}{4}\right\}$$

$$a_8 = -g\left(\theta\Delta t\right)^2\left\{\left(l_{\xi\xi}D\right)_{j+1/2,k} + \frac{\left(l_{\xi\eta}D\right)_{j,k+1/2} - \left(l_{\xi\eta}D\right)_{j,k-1/2}}{4}\right\}$$

$$a_5 = 1 + g\left(\theta\Delta t\right)^2\left\{\left(l_{\xi\xi}D\right)_{j-1/2,k} + \left(l_{\xi\xi}D\right)_{j+1/2,k} + \left(l_{\eta\eta}D\right)_{j,k-1/2} + \left(l_{\eta\eta}D\right)_{j,k+1/2}\right\}$$

Assembly of equation (2.8.20) over the entire surface domain creates a matrix with nine diagonal nonzero terms. The matrix can then be solved to obtain the solution for the value of free surface elevation at the current time level. To aid in solving the resulting matrix, a preconditioned biconjugate gradient method is used, with the preconditioner being the diagonal elements of the matrix [99]. The solution of the matrix will then be used to update the grid metric terms and as well as the approximate velocities. The subsequent step will be to obtain the dynamic pressure terms for the equations and update all velocities such that mass balance can be satisfied.

2.9 - *Pressure Poisson Equation*

The computation of the dynamic pressure terms is rather similar to that of the free surface terms outlined earlier. In the discussions thus far, the dynamic pressures were omitted from the momentum equation when calculating the free surface terms. If dynamic pressure is not desired (*i.e.*, hydrostatic problems), then the velocities at $n+1$ time level may simply be calculated based on the incompressibility constraint as mentioned earlier. However, if dynamic pressure is to be considered, corrections have to be made to the velocities, to include the dynamic pressures into the Navier-Stokes

equations, in order to satisfy the incompressibility constraint. This requires the solution of the Pressure Poisson Equation (PPE). Similar to the free surface calculation procedure, the dynamic pressure calculation requires intermediate information, specifically the approximate velocities. The approximate velocities are updated to the *n+1* time level after the solution of the dynamic pressure is obtained. The relationships are

$$\begin{aligned} u_\xi^{n+1} &= \tilde{u}_\xi^{n+1} - \theta\Delta t P_\xi^{n+1} - (1-\theta)\Delta t P_\xi^{n} \\ u_\eta^{n+1} &= \tilde{u}_\eta^{n+1} - \theta\Delta t P_\eta^{n+1} - (1-\theta)\Delta t P_\eta^{n} \\ u_\zeta^{n+1} &= \tilde{u}_\zeta^{n+1} - \theta\Delta t P_\zeta^{n+1} - (1-\theta)\Delta t P_\zeta^{n} \end{aligned} \tag{2.9.1}$$

Since this procedure is the last step to compute and update the final velocities for a full temporal iteration, the solution should be directly related to the continuity equation to preserve mass. u_i^{n+1} has to satisfy the continuity equation, and therefore at the *n+1* time level this relationship must hold true

$$u_{\xi\, j+1/2,k,l}^{n+1} - u_{\xi\, j-1/2,k,l}^{n+1} + u_{\eta\, j,k+1/2,l}^{n+1} - u_{\eta\, j,k-1/2,l}^{n+1} + u_{\zeta\, j,k,l+1/2}^{n+1} - u_{\zeta\, j,k,l-1/2}^{n+1} = 0 \tag{2.9.2}$$

Substituting u_i^{n+1} from equation (2.9.1) into the discretized continuity equation, equation (2.9.2), resulted in

$$\begin{aligned} &\left(\tilde{u}_\xi^{n+1} - \theta\Delta t P_\xi^{n+1} - (1-\theta)\Delta t P_\xi^{n}\right)_{j+1/2,k,l} - \left(\tilde{u}_\xi^{n+1} - \theta\Delta t P_\xi^{n+1} - (1-\theta)\Delta t P_\xi^{n}\right)_{j-1/2,k,l} + \\ &\left(\tilde{u}_\eta^{n+1} - \theta\Delta t P_\eta^{n+1} - (1-\theta)\Delta t P_\eta^{n}\right)_{j,k+1/2,l} - \left(\tilde{u}_\eta^{n+1} - \theta\Delta t P_\eta^{n+1} - (1-\theta)\Delta t P_\eta^{n}\right)_{j,k-1/2,l} + \\ &\left(\tilde{u}_\zeta^{n+1} - \theta\Delta t P_\zeta^{n+1} - (1-\theta)\Delta t P_\zeta^{n}\right)_{j,k,l+1/2} - \left(\tilde{u}_\zeta^{n+1} - \theta\Delta t P_\zeta^{n+1} - (1-\theta)\Delta t P_\zeta^{n}\right)_{j,k,l-1/2} = 0 \end{aligned} \tag{2.9.3}$$

Rearranging equation (2.9.3) yields,

$$\begin{aligned} &\left(\theta\Delta t P_\xi^{n+1}\right)_{j+1/2,k,l} - \left(\theta\Delta t P_\xi^{n+1}\right)_{j-1/2,k,l} + \left(\theta\Delta t P_\eta^{n+1}\right)_{j,k+1/2,l} - \left(\theta\Delta t P_\eta^{n+1}\right)_{j,k-1/2,l} + \\ &\left(\theta\Delta t P_\zeta^{n+1}\right)_{j,k,l+1/2} - \left(\theta\Delta t P_\zeta^{n+1}\right)_{j,k,l-1/2} = \\ &\left(\tilde{u}_\xi^{n+1} - (1-\theta)\Delta t P_\xi^{n}\right)_{j+1/2,k,l} - \left(\tilde{u}_\xi^{n+1} - (1-\theta)\Delta t P_\xi^{n}\right)_{j-1/2,k,l} + \\ &\left(\tilde{u}_\eta^{n+1} - (1-\theta)\Delta t P_\eta^{n}\right)_{j,k+1/2,l} - \left(\tilde{u}_\eta^{n+1} - (1-\theta)\Delta t P_\eta^{n}\right)_{j,k-1/2,l} + \\ &\left(\tilde{u}_\zeta^{n+1} - (1-\theta)\Delta t P_\zeta^{n}\right)_{j,k,l+1/2} - \left(\tilde{u}_\zeta^{n+1} - (1-\theta)\Delta t P_\zeta^{n}\right)_{j,k,l-1/2} \end{aligned} \tag{2.9.4}$$

The left hand side of equation (2.9.4) contains the unknown variables while the right hand side of the equation contains the known variables. For simplicity, the notation *RHS* will be used to denote the right hand side components of equation (2.9.4). Rewriting equation (2.9.4) gives

$$\left(\theta\Delta tP_{\xi}^{\,n+1}\right)_{j+1/2,k,l}-\left(\theta\Delta tP_{\xi}^{\,n+1}\right)_{j-1/2,k,l}+\left(\theta\Delta tP_{\eta}^{\,n+1}\right)_{j,k+1/2,l}-\left(\theta\Delta tP_{\eta}^{\,n+1}\right)_{j,k-1/2,l}+$$
$$\left(\theta\Delta tP_{\zeta}^{\,n+1}\right)_{j,k,l+1/2}-\left(\theta\Delta tP_{\zeta}^{\,n+1}\right)_{j,k,l-1/2}=RHS \tag{2.9.5}$$

The equation contains the dynamic pressure gradient terms, denoted by P and is centrally discretized into a more computable form. This operation yields

$$b_1p_{j-1,k,l-1}^{n+1}+b_2p_{j,k-1,l-1}^{n+1}+b_3p_{j,k,l-1}^{n+1}+b_4p_{j,k+1,l-1}^{n+1}+b_5p_{j+1,k,l-1}^{n+1}+b_6p_{j-1,k-1,l}^{n+1}+$$
$$b_7p_{j-1,k,l}^{n+1}+b_8p_{j-1,k+1,l}^{n+1}+b_9p_{j,k-1,l}^{n+1}+b_{10}p_{j,k,l}^{n+1}+b_{11}p_{j,k+1,l}^{n+1}+b_{12}p_{j+1,k-1,l}^{n+1}+b_{13}p_{j+1,k,l}^{n+1}+ \tag{2.9.6}$$
$$b_{14}p_{j+1,k+1,l}^{n+1}+b_{15}p_{j-1,k,l+1}^{n+1}+b_{16}p_{j,k-1,l+1}^{n+1}+b_{17}p_{j,k,l+1}^{n+1}+b_{18}p_{j,k+1,l+1}^{n+1}+b_{19}p_{j+1,k,l+1}^{n+1}=RHS$$

where,

$$b_1=-\theta\Delta t\frac{l_{\xi\zeta(j-1/2,k,l)}+l_{\xi\zeta(j,k,l-1/2)}}{4},\ b_2=-\theta\Delta t\frac{l_{\eta\zeta(j,k-1/2,l)}+l_{\eta\zeta(j,k,l-1/2)}}{4}$$

$$b_4=\theta\Delta t\frac{l_{\eta\zeta(j,k+1/2,l)}+l_{\eta\zeta(j,k,l-1/2)}}{4},\ b_5=\theta\Delta t\frac{l_{\xi\eta(j+1/2,k,l)}+l_{\xi\eta(j,k,l-1/2)}}{4}$$

$$b_6=-\theta\Delta t\frac{l_{\xi\eta(j-1/2,k,l)}+l_{\xi\eta(j,k-1/2,l)}}{4},\ b_8=\theta\Delta t\frac{l_{\xi\eta(j-1/2,k,l)}+l_{\xi\eta(j,k+1/2,l)}}{4}$$

$$b_{12}=\theta\Delta t\frac{l_{\xi\eta(j+1/2,k,l)}+l_{\xi\eta(j,k-1/2,l)}}{4},\ b_{14}=-\theta\Delta t\frac{l_{\xi\eta(j+1/2,k,l)}+l_{\xi\eta(j,k+1/2,l)}}{4}$$

$$b_{15}=\theta\Delta t\frac{l_{\xi\zeta(j-1/2,k,l)}+l_{\xi\zeta(j,k,l+1/2)}}{4},\ b_{16}=\theta\Delta t\frac{l_{\eta\zeta(j,k-1/2,l)}+l_{\eta\zeta(j,k,l+1/2)}}{4}$$

$$b_{18}=-\theta\Delta t\frac{l_{\eta\zeta(j,k+1/2,l)}+l_{\eta\zeta(j,k,l+1/2)}}{4},\ b_{19}=-\theta\Delta t\frac{l_{\xi\zeta(j+1/2,k,l)}+l_{\xi\zeta(j,k,l+1/2)}}{4}$$

$$b_3=-\theta\Delta t\left(l_{\zeta\zeta(j,k,l-1/2)}+\frac{l_{\xi\zeta(j-1/2,k,l)}-l_{\xi\zeta(j+1/2,k,l)}+l_{\eta\zeta(j,k-1/2,l)}-l_{\eta\zeta(j,k+1/2,l)}}{4}\right)$$

$$b_7=-\theta\Delta t\left(l_{\xi\xi(j-1/2,k,l)}+\frac{l_{\xi\eta(j,k-1/2,l)}-l_{\xi\eta(j,k+1/2,l)}+l_{\xi\zeta(j,k,l-1/2)}-l_{\xi\zeta(j,k,l+1/2)}}{4}\right)$$

$$b_9=-\theta\Delta t\left(l_{\eta\eta(j,k-1/2,l)}+\frac{l_{\xi\eta(j-1/2,k,l)}-l_{\xi\eta(j+1/2,k,l)}+l_{\eta\zeta(j,k,l-1/2)}-l_{\eta\zeta(j,k,l+1/2)}}{4}\right)$$

$$b_{11}=-\theta\Delta t\left(l_{\eta\eta(j,k+1/2,l)}+\frac{l_{\xi\eta(j+1/2,k,l)}-l_{\xi\eta(j-1/2,k,l)}+l_{\eta\zeta(j,k,l+1/2)}-l_{\eta\zeta(j,k,l-1/2)}}{4}\right)$$

$$b_{13}=-\theta\Delta t\left(l_{\xi\xi(j+1/2,k,l)}+\frac{l_{\xi\eta(j,k+1/2,l)}-l_{\xi\eta(j,k-1/2,l)}+l_{\xi\zeta(j,k,l+1/2)}-l_{\xi\zeta(j,k,l-1/2)}}{4}\right)$$

$$b_{17} = -\theta\Delta t\left(l_{\zeta\zeta(j,k,l+1/2)} + \frac{l_{\xi\zeta(j+1/2,k,l)} - l_{\xi\zeta(j-1/2,k,l)} + l_{\eta\zeta(j,k+1/2,l)} - l_{\eta\zeta(j,k-1/2,l)}}{4} \right)$$

$$b_{10} = \theta\Delta t\left(l_{\xi\xi(j-1/2,k,l)} + l_{\xi\xi(j+1/2,k,l)} + l_{\eta\eta(j,k-1/2,l)} + l_{\eta\eta(j,k+1/2,l)} + l_{\zeta\zeta(j,k,l-1/2)} + l_{\zeta\zeta(j,k,l+1/2)} \right)$$

The sets of equations were solved using the same technique as the free surface equation; however, with more involved computational effort due to the greater number of unknowns. The assembly of equation (2.9.6) over the entire domain yields a nineteen-diagonal system of equations. If the horizontal grid is orthogonal to the vertical grid, the matrix can be reduced into a seven-diagonal form. The preconditioned biconjugate gradient is again used to solve for the dynamic pressures.

In some simulations where the pressure converges very slowly, it is best to provide a Dirichlet boundary condition to the pressure. This pins the pressure at a point or cell in the domain, helping to accelerate the pressure iterations. The pressure at any single point is not as important as the pressure gradients to solve the ensemble averaged Navier-Stokes equations. Therefore, an arbitrary assignment of pressure at a single node/cell will not affect the solution of the Navier-Stokes equations. This is especially true in closed conduit flow simulation. In open channel flows, the free surface is an essential part of the computation and the pressure boundary condition at the free surface has to be zero (atmospheric pressure), by definition.

Once the solution to the PPE is obtained, the determination of the hydrodynamics in the Navier-Stokes equations is complete for one time step. The numerical procedure can then proceed to solve the scalar transport equation.

2.10 - *Transport Equation for Air Bubbles*

Air entrainment in the form of small spherical air bubbles is considered in this model and is the primary focus of this work. Small air bubbles can be modeled as scalars and transported via fluid convection. Unlike the Lagrangian family of tracking techniques, the proposed method does not track individual or interfaces of air bubbles, but reports the concentration of air bubbles for each computation cell in the domain. This procedure of tracking air bubble concentrations require far less memory and arithmetic effort

compared to the Lagrangian techniques, which demands tracking precise location of individual air bubble. It is not the intent of this work to be able to track every individual air bubbles and its position. Tracking the concentration of air bubbles will be adequate for the application of pump station design and modeling. The typical scalar transport equation will be used to track air concentration. The typical scalar transport equation is written as

$$\frac{\partial \phi}{\partial t}+\frac{\partial u\phi}{\partial x}+\frac{\partial v\phi}{\partial y}+\frac{\partial w\phi}{\partial z}-D_T\left(\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\phi}{\partial y^2}+\frac{\partial^2\phi}{\partial z^2}\right)=s_\phi \tag{2.10.1}$$

where, ϕ are the scalar quantities being transported,

s_ϕ is the source terms of the scalar quantities, and

D_T is the eddy diffusivities

Experimental studies from different authors suggested that the diffusivity of air bubbles is in the same order of magnitude as the eddy viscosity of the fluid ([16], [23], [67]). It is assumed that the diffusivity for air bubbles will take on the positive values of turbulent eddy viscosities. In generalized coordinate system, the equation has the form

$$\frac{\partial \phi}{\partial t}+\frac{\partial fc_\xi}{\partial \xi}+\frac{\partial fc_\eta}{\partial \eta}+\frac{\partial fc_\zeta}{\partial \zeta}=s_\phi \tag{2.10.2}$$

with the variable *fc* being,

$$\begin{aligned}
fc_\xi &= u_\xi\phi - D_T\left(l_{\xi\xi}\frac{\partial\phi}{\partial\xi}+l_{\xi\eta}\frac{\partial\phi}{\partial\eta}+l_{\xi\zeta}\frac{\partial\phi}{\partial\zeta}\right)\\
fc_\eta &= u_\eta\phi - D_T\left(l_{\xi\eta}\frac{\partial\phi}{\partial\xi}+l_{\eta\eta}\frac{\partial\phi}{\partial\eta}+l_{\eta\zeta}\frac{\partial\phi}{\partial\zeta}\right)\\
fc_\zeta &= (u_\eta\phi-\frac{dz}{dt}\zeta_z) - D_T\left(l_{\xi\eta}\frac{\partial\phi}{\partial\xi}+l_{\eta\eta}\frac{\partial\phi}{\partial\eta}+l_{\eta\zeta}\frac{\partial\phi}{\partial\zeta}-\zeta_z^2\frac{\partial\phi}{\partial\zeta}\right)
\end{aligned} \tag{2.10.3}$$

The grid metric terms are the same as defined in equation (2.6.3) through (2.6.5). Temporal discretization of equation (2.10.3) takes on the same predictor-corrector scheme used for the LES momentum equations. For the sake of completeness, the computation of the predictor and corrector steps are shown again and are as follows

$$\phi^{n+\frac{1}{2}}=\phi^n-\frac{\Delta t}{2}\left(\frac{\partial u_\xi\phi}{\partial\xi}+\frac{\partial u_\eta\phi}{\partial\eta}+\frac{\partial u_\zeta\phi}{\partial\zeta}-\frac{\partial\phi}{\partial\zeta}\frac{dz}{dt}\zeta-s_\phi\right)^n \tag{2.10.4}$$

$$\phi^{n+1} = \phi^n - \Delta t\left(\frac{\partial u_\xi \phi}{\partial \xi} + \frac{\partial u_\eta \phi}{\partial \eta} + \frac{\partial u_\zeta \phi}{\partial \zeta} - \frac{\partial \phi}{\partial \zeta}\frac{dz}{dt}\zeta - s_\phi\right)^{n+\frac{1}{2}} \tag{2.10.5}$$

As indicated, the procedure for solving the scalar transport equation is similar to that of the predictor step in the momentum equation. The scalar quantities at the *n+1/2* time level will be computed from the known values at the *n* time level. The corrector step will then proceed, using the known scalar quantities at the *n+1/2* time level.

2.11 - *Extrapolation using Monotone Upstream Scheme for Conservation Law (MUSCL scheme)*

The treatment of the advection terms requires special attention to avoid oscillations when computing surface with adverse flux gradients. The Monotone Upstream Scheme for Conservation Law (MUSCL) is used to treat the advection fluxes. This scheme provides a second order accurate solution even for problems which exhibit shocks, discontinuities, or large gradients [120] and is one of the most widely used scheme for capturing shock in compressible flows [33]. The successful application of the MUSCL scheme have been documented for *2*-dimensional [27] and *3*-dimensional [79] problems.

To compute an advection flux, the MUSCL procedure is used to extrapolate piecewise linear approximation of flow variable to the cell faces making use of the gradients to the left and right of the cell face. Bradford and Katopodes [14] described the MUSCL scheme advection flux equation as

$$f = u_\perp \psi = \frac{1}{2}\left(u_\perp\left(\psi_L + \psi_R\right) - \left|u_\perp\right|\left(\psi_R - \psi_L\right)\right) \tag{2.11.1}$$

The notation ψ is a generic representation of an advected flow variable and $u_\perp$ is used to represent the velocity component perpendicular to the cell face in consideration. The subscripts *L* and *R* are the reconstructed or extrapolated quantities on the left and right side of a cell face. The extrapolation enables the evaluation of flow variables in a staggered grid at intended cell faces, which are used to calculate the advection fluxes. In equation (2.11.1) ψ_L *and* ψ_R are defined as

$$\psi_R = \psi_{j+1,k,l} - \frac{1}{2}\overline{\Delta\psi}_{\xi j+1,k,l} \tag{2.11.2}$$

$$\psi_L = \psi_{j,k,l} + \frac{1}{2}\overline{\Delta\psi}_{\xi j,k,l} \tag{2.11.3}$$

The figure that follows illustrate the boundary of cell with indices *j,k,l* and how a flow variable can be calculated on the cell face labeled *E* in the ξ direction. The variables on the *E* face of the cell can be constructed from two different directions, the upward extrapolation from cell *j,k,l*, which will produce ψ_L and the downward extrapolation from cell *j+1,k,l*, which gives ψ_R.

In order to calculate ψ_L *and* ψ_R, the gradients from both sides have to also be calculated. An upwind discretization scheme is used to evaluate the average gradients, $\overline{\Delta\psi}_\xi$. This technique was pioneered by Van Leer [120] in 1979. The average gradients can be calculated in different ways, and in the scope of this work, the beta family of averages is used. The criteria for calculating $\overline{\Delta\psi}_\xi$ is

$$\overline{\Delta\psi} = \begin{Bmatrix} \text{sign(a)}\min\left(\max\left(|a|,|b|\right), \beta\min\left(|a||b|\right)\right) & ab > 0 \\ 0 & ab \le 0 \end{Bmatrix} \tag{2.11.4}$$

where $1 \le \beta \le 2$, $a = \psi_{j,k} - \psi_{j-1,k}$ and $b = \psi_{j+1,k} - \psi_{j,k}$. Setting β to *2* yields a less dissipative Superbee average, while if the values of β approaches *1*, solution obtain will have more numerical dissipation. In this work, β is chosen to be *2*. Although the procedure shown above is only for a single direction, the MUSCL scheme can easily be extended to the other directions with the same outlined procedure.

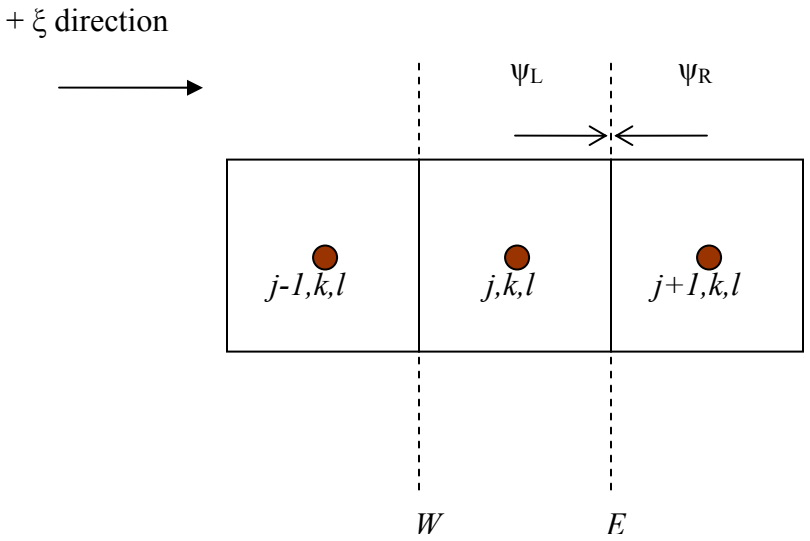

Figure 2-2: MUSCL extrapolation scheme for advection flux in an arbitrary axis.

2.12 - *General Boundary Conditions*

In numerical modeling, simulations are performed for a finite spatial region or domain size. However, environments outside of the domain of interest have to be considered as changes in the outside environment affect the final solution of the system. Boundary conditions are used as constraints to artificially truncate the region of interest in such a way that the fluid behavior outside the region of interest can be described correctly by the prescribed boundary conditions. It should be emphasized that boundary conditions are very important as they can affect the stability and accuracy of a numerical scheme [34], and consequently the validity of the solution if not implemented correctly. A chosen boundary condition should not introduce more additional unknowns and instead, the applied boundary condition should assist in solving the problem by either extrapolation or one sided difference [54].

The present application considered the use of two main types of boundaries, namely the Dirichlet type (specified) boundary conditions and the Neumann type boundary conditions. The Dirichlet boundary condition is one of the simplest types of boundary condition. Dirichlet boundary requires the specification of values on the boundary plane, and therefore conditions on the boundary are fixed regardless of changes in the domain. A Neumann boundary is a fixed flux boundary, and thus is a derivative type boundary. The Neumann boundary conditions make use of a specified gradient value to obtain solution at a particular boundary.

In this work, the main boundaries are the inflow, outflow, side boundaries and bottom boundary. Flow simulations require different boundaries conditions for different sides of the domain. Prescribing wrong boundary conditions may result in erroneous solutions. Therefore, it was essential to understand the different boundary conditions in order to appropriately choose the correct boundary for its respective application. Diffusive fluxes at the boundaries are intentionally left out of the discussion because the influences of these terms are far less compared to the convective terms.

Inflow boundary Conditions

The inflow boundary condition is best specified based on results obtained from a previous simulation [54]. A good inflow condition contains complete velocity field at the inflow section with all its fluctuating components at each time step. Dirichlet or Neumann type boundary condition can be used for the inflow boundary. For homogeneous isotropic turbulence, the periodic boundary condition can be applied to the inflow boundary. The periodic boundary condition basically applies the result at the outflow boundary to the inflow boundary, creating a looping effect to the domain. In some sense, the periodic boundary condition is similar to the Dirichlet boundary condition in which the velocities at the inflow boundary is supplied with known values, but the boundary information changes at each time step.

A specified velocity boundary is primarily used at the inlet boundary of a domain. At the inlet, information of the incoming flows are passed on from this boundary to the interior nodes. Knowledge on the behavior of the inlet velocities profile is needed in order to specify such boundary condition. One important aspect of the inlet specified velocity boundary condition is that the inlet velocities are the main driving forces to the domain, especially for convection dominated flow. The flux generated by the inlet velocities will be convected downstream and thus influence the temporal and spatial solution in the domain.

In a scenario where the downstream condition influences the upstream boundary, a specified inflow velocity profile may not be appropriate as this will physically mean that the flowrate of the problem changes as depth changes. Therefore a Neumann type boundary can be applied to prescribe the flux across a boundary. By using the Neumann boundary, the flowrate can be maintained whenever adjustment in depths occurs.

The only disadvantage of using the Dirichlet and Neumann boundaries for the inflow condition is that for channel flow simulation, the inflow condition must contain turbulent fluctuation components. Providing random fluctuations to the velocities component is not sufficient as the poor temporal and spatial correlation of the velocities fluctuation will be damped out very rapidly, as will be explained in section 2.14. It is up to the researcher to decide on the method to specify the individual velocities fluctuations, be it statistical

randomly generated fluctuations, quasi-random, or empirical data from experimentation. However, the object is still to describe the inflow velocities such that the system can be described completely and correctly.

Outlet boundary conditions

At the outflow boundary several different combination of boundary condition can be used. A typical choice for the outlet boundary condition of a channel flow simulation is the constant flux boundary condition, a Neumann type boundary condition. The boundary can be described as

$$\frac{\partial \alpha_l}{\partial x_l} = \text{constant}$$

The variable α_i denotes a generic variable to represent either a scalar or vector quantity in its coordinate direction. To use this boundary condition, flux across the boundary has to be prescribed. In simple incompressible steady flow problems, the constant is simply set to zero. This extrapolation may not hold true for unsteady flows with shocks. It is best to understand the flow in the domain of interest before implementing this boundary condition, particularly in determining the value of the flux. For the unsteady flow condition described, the use of the unsteady convective boundary can be used [54]. The convective boundary condition is sometimes known as the radiation or Orlanski boundary condition [97]. The Orlanski boundary condition has the form

$$\frac{\partial \alpha}{\partial t} + C_\alpha \frac{\partial \alpha}{\partial x_i} = 0$$

The basic equation of the Orlanski boundary is also similar to that of the Sommerfeld radiation condition, with the exception that C_α be treated as a propagation velocity which is a function of its neighboring grid points. However, depending on the application of this boundary, C_α may be substituted with the averaged normal velocity across the exit boundary ([60] , [115]) or simply a velocity that is independent of the outflow plane but still maintain the overall conservation [54].

Although outflow boundary conditions can take several forms, the objective of this boundary condition remains the same - to minimize or eliminate the reflection of waves at the boundary and provide a "realistic" truncation of the actual domain.

No-slip boundary

The no-slip boundary condition is a Dirichlet type boundary condition. The no-slip boundary is used as physical boundary for a solid wall. In theory, all velocities on the no-slip boundary must be equals to zero (or the velocity of a moving wall)

$$u_i = 0$$

The use of the no-slip boundary is straight forward. Mass and momentum transfer across the wall boundary is forbidden. Although simple, the no-slip boundary does not describe the material property or the roughness associated with the surface of a boundary, unless detailed bathymetry profile is provided, which include the protruding roughness elements. As far as surface roughness is concerned, the no-slip boundary assumes a smooth surface. Whenever the no-slip boundary condition is applied, the chosen grid size must be very small in order to resolve the tangential velocities adjacent to this boundary where the velocities are rapidly changing [127] and the velocity gradients are very large. Unless the flow is a laminar type flow, the use of no-slip boundary is computationally demanding and therefore is limited.

Free slip boundary

The free-slip boundary is used to describe the continuation of a domain which is infinite in length. The use of free slip condition assumes symmetry of the flow for planes parallel to the boundary with a zero convective flux, for the direction perpendicular to the boundary. In simulations of *2*-dimensional problems with a *3*-dimensional model, the free slip boundary is often used for the direction that is of least important. In some instances, this boundary condition can be used to describe an infinitely deep domain, where the conditions outside the domain will not have significant influence on the flow inside the

domain. An example of this scenario is the free stream region of a flow. In this work, the free slip boundary is often used to truncate the flow domain to a manageable length, assuming symmetry in the direction of the truncation.

Free surface boundary condition

Calculation of the free surface equation should be accompanied by the necessary boundary conditions. The specified surface elevation boundary is nothing more than restricting the free-surface movement at the boundary. More than likely, the specified surface elevation boundary condition will be coupled with the specified velocity boundary condition to re-create the inlet at a controlled section, such as the sluice gate. In supercritical flows where the upstream boundary acts as the control, the downstream free surface boundary has to be an open boundary where the gradient of the surface elevation is assumed to be zero, or simply computed using the method of characteristics.

For subcritical flows, the fixed surface elevation boundary condition is used for downstream control and this boundary can be coupled with a fixed flux boundary. The upstream boundary can be of a fixed velocity or flux, and the free surface adjustment will be calculated based on the changes of flux throughout the domain.

2.13 - *Near Wall Treatment*

Flow domain with solid wall boundaries often poses a problem when resolving the velocities near the wall. This is especially clear when evaluating the gradient terms, as the near wall velocity gradients are often very large. In practice, there are two options available to obtain the velocity profile near the wall [4], namely:

1. The use of a wall function
2. Grid refinement method

The later option requires that additional grids to be specified near the boundary to resolve the velocity profile in the inner region. Grid refinement requires more nodes to be added near and to the wall, consequently increasing computation effort. Typically, this approach

is taken when simulations are done in a very fine grid size, like performing a DNS simulation, or using non-uniform grid in the direction perpendicular to the wall. The reason for using such fine grids near the wall is to break down the large velocity gradients into manageable magnitude, thus avoiding instability.

The use of wall function involves specifying an empirical function previously obtained, *e.g.* the law of the wall or the power law function to regions bounded by the boundary layer. This method is especially popular for simulation of high Reynolds number flows [94]. The law of the wall has proved to be a popular choice. Wall functions are convenient to use and do not require intensive arithmetic effort. Using the wall function relates to either the velocities or shear in the boundary layer. An example of the wall function is [127]

$$u^+ = \frac{1}{\kappa}\ln\left(\frac{z}{z^+}\right) \tag{2.13.1}$$

Equation (2.13.1) is formally known as the law of the wall function for fully rough boundaries. The main idea of using the law of the wall function is to truncate the near wall region, and hence no need for very fine grid resolution. Several other forms of the wall function can be found and examples of these wall functions are the power law ([1], [68]) and the quadratic drag law function ([48], [72], [78]).

$$u = U_\infty\left(\frac{z}{\delta}\right)^{\frac{1}{n}} \tag{2.13.2}$$

The former example, shown as equation (2.13.2), can be used analogous to the log law function by providing a velocity at the boundary, while the later example, shown as equation (2.13.7) uses the drag coefficient to obtain the shear across the boundary. The power law function was not used in this work and further elaboration is unnecessary. The quadratic drag law wall function and its implementation will be further elaborated later in this section.

The idea of using the partial slip boundaries or wall functions is to provide a "fictitious" boundary directly above the physical boundary so that the tangential velocities at this boundary are not equal to zero, therefore avoiding the calculation of the high velocity gradients explained earlier. This requires that a slip velocity or the shear

stress be calculated at the boundary to account for the truncated physical boundary. The shear stress can be related to a shear velocity, u_τ, [17], namely

$$u_\tau^2 = \frac{\tau_w}{\rho} = \nu \left.\frac{\partial u}{\partial z}\right|_w \tag{2.13.3}$$

To make use of equation (2.13.3), the gradient of the normal velocities has to be known. The direct application of this equation will introduce difficulties because to calculate the shear stress, the velocity gradient needs to be calculated near the wall. Linear interpolation of the velocity profile at the boundary is erroneous in approximating the gradient of the log profile. Thus, another approach had to be taken to calculate the slip velocity for better approximation of the gradient. The slip velocity can be calculated with the following relationships [127]

$$u_\tau{}^2 = \frac{u^2{}_{mag}}{A_m} \tag{2.13.4}$$

$$A_m = \frac{1}{\kappa^2}\ln^2\left(\frac{z'}{z^+}\right) \tag{2.13.5}$$

$$u^2{}_{mag} = \sum_{i=1, i\neq j}^{N_{sd}} u_i{}^2(z') \tag{2.13.6}$$

Where z' is the distance of the first grid location away from the wall, κ is the Von Karmon constant , typically set to the value of *0.41,* and z^+ can be generalized to [127]

$$z^+ = \frac{k_s}{30}$$

for a fully turbulent flow over a rough bed. The variable k_s is also known as the equivalent roughness. Shear at the wall can then be calculated based on equation (2.13.3). However, since the discretization of the governing equations produced a staggered grid system, and in addition, the implicit treatment of the vertical diffusion terms requires that velocities be solved simultaneously, equation (2.13.4) through (2.13.6) cannot be used. Another approach must be taken to apply the wall function as the bottom boundary condition, that could provide a wall shear stress for the direct calculation in the implicit iteration procedure of the vertical diffusion terms.

In this work, the quadratic drag law was used to approximate the bottom wall shear. The quadratic drag law is often used in numerical models ([9], [48], [72], [78]). The shear stress at the wall is written as

$$\tau_w = C_{dw}|u_i|u_i \tag{2.13.7}$$

where, C_{dw} is the wall drag coefficient.

The wall drag coefficient can be calculated from knowing the distance of the first grid away from the wall and the wall roughness, given as [48]

$$C_{dw} = \frac{\kappa^2}{\left(\ln(\frac{30z}{k_s})\right)} \tag{2.13.8}$$

The implementation of equation (2.13.7) and (2.13.8) provides a wall shear stress which can then be used as part of the calculation involved in the Navier-Stokes equations. Equation (2.13.7) and (2.13.8) were employed as a wall model to provide shear as a boundary condition as needed. This eliminated the need to resolve the problem in the vertical direction to the scale of less than the laminar sublayer. Since the solution of the implicit procedure required that the velocities be solved simultaneously, the quadratic drag law was the appropriate choice of wall model for this work. The use of this wall model eliminated the need for a slip velocity to be calculated at the wall boundary, and instead provided a shear based on the velocity of the first grid above the wall. The shear was then associated with vertical diffusion terms to be solve simultaneously. To illustrate the implementation of the wall boundary, the LES equations will only be written for the vertical diffusion component because of the implicit treatment of this term. For simplicity, only one direction, ξ, will be shown, and is given as

$$\frac{u_\xi^{n+1} - u_\xi^n}{\Delta t} - \theta\left(w_d \frac{\partial u_\xi}{\partial \zeta} + \frac{\partial m_\xi}{\partial \zeta}\right) = RHS \tag{2.13.9}$$

Further discretization of this equation will give a tridiagonal matrix system relating the terms $u_{\xi,l-1}$, $u_{\xi,l}$ and $u_{\xi,l+1}$ shown below as

$$u_\xi - \theta\Delta t\left(w_d\left[u_{\xi,l+1} - u_{\xi,l}\right] + \nu_T\zeta^2\left[u_{\xi,l-1} - 2u_{\xi,l} + u_{\xi,l+1}\right]\right) = RHS \tag{2.13.10}$$

Following that, the coefficients in the matrix are

$$a_l = -\theta\Delta t\left[\nu_{T,l-\frac{1}{2}} - w_d\right] \quad , \quad a_u = -\theta\Delta t\left[\nu_{T,l+\frac{1}{2}} + w_d\right] \quad , \quad a_d = 1 - a_l - a_u$$

Where a_l is the lower diagonal coefficient, a_u being the upper diagonal coefficient and a_d is the diagonal coefficient in the matrix. The wall boundary, can be applied as follows

$$a_d = a_d + a_l\left(1 - wbc\right) \tag{2.13.11}$$

where the term *(1-wbc)* is added to the diagonal term of the first row in the matrix, where the wall function is applied. The term *wbc* is obtained by equating the basic definition of the wall shear stresses, equation (2.13.3) with that of the quadratic drag law, equation (2.13.7). The relationship is shown as

$$\frac{\partial u}{\partial z}\nu = C_{dw}\left|u\right|u = \tau_w \tag{2.13.12}$$

discretizing the equation will yield

$$\frac{u_l - u_{l-1}}{\Delta z}\nu = C_{dw}\left|u_l\right|u_l \tag{2.13.13}$$

rearranging this equation gives

$$u_{l-1} = u_l - \frac{C_{dw}\left|u_l\right|u_l\Delta z}{\nu} = u_l\left(1 - bcw\right) \tag{2.13.14}$$

Substituting equation (2.13.14) into equation (2.13.10) gives the diagonal terms as described in equation (2.13.11). The term *bcw* for a no-slip condition is *2* while for a free slip boundary, *bcw* is set to *1*.

The disadvantage of using the partial slip boundary is that the computation domain lacks the real characteristics that define the energy dissipation near the wall. This weakness is apparent when the dynamic SGS is used to model the turbulent dissipation. Absence of the correct boundary information from the wall will lead to inaccurate closure for the dynamic SGS model. Arguments can be made that the shear stress or partial slip velocity provided will be sufficient for the subgrid scale modeling. This argument can only be true when the standard Smagorinsky model is used as the SGS model. The dynamic SGS model takes into consideration the velocities components at the grid and test filter levels, and thus requires these information at the wall boundaries as well. Unless correct and accurate information accompanies the partial slip wall boundary condition, the dynamic SGS model might not be able to capture the correct small scale

energy dissipation generated near the wall. This deficiency had been recognized and attempts to improve on the near wall SGS modeling are still in progress ([17], [89]). The simplest, but not necessarily the best way, to overcome this deficiency is to employ the simple mixing length eddy viscosity model with an ad-hoc wall damping function ([17], [98], [123]). The direct calculation of the eddy viscosity is as simple as applying the equation found in [17]

$$\begin{aligned} \nu_t &= (\kappa y)^2 \left|\tilde{S}\right| D_f \\ D_f &= [1 - \mathrm{e}^{-\left(\frac{y^+}{A^+}\right)^3}] \end{aligned} \tag{2.13.15}$$

More sophisticated methods may involve the use of nested grids [17] along the wall or overlaying finer grid zone into the first computational cell (also known as the Two-Layer Model, [98]). Then the Reynolds stresses are directly computed, though inaccuracy may arise in the nested grid level.

2.14 - *Turbulent Inflow Generation*

Turbulent inflow specification is another challenge and is still actively being researched ([40], [41], [42], [45], [73], [84]). In the case of laminar flows, inflow boundary condition can be specified as just the mean velocity profile. Laminar flows are viscous forces dominated flow and thus, diffusion is controlled only by the property of the working fluid, namely the molecular viscosity. The velocity fluctuations in laminar flows are negligible. Contrary to the laminar flows, the nature of turbulent flows is different in terms of energy production and dissipation. In turbulent flows, energy is dissipated or gained (backscattering) not only at molecular level, but also by the large and small eddies produced from the velocities fluctuations. Unless a 'long' domain is provided to allow the flow to evolve from a laminar flow to a turbulent flow, specifying a mean velocity profile is simply insufficient to describe the turbulent properties at the inflow boundary.

A realistic inflow boundary condition for turbulent flows includes the mean velocities profile plus the respective velocity fluctuations of each component, as described in the

LES decomposition, shown as equation (2.3.1). Intuitively, the simplest way to prescribe the velocity fluctuations are to randomly generate numbers and scale the numbers to the magnitude desired, and repeating this process for the inflow boundary for each time step. The velocities fluctuations generated using this method are also known as 'white noise' fluctuations [77]. Generating 'white noise' fluctuations is not the correct approach to describe turbulence, as there are essentially little to none spatial or temporal correlations in the velocities fluctuations. Or simply put, not enough energy containing eddies at the low wave number range. The pseudo turbulence will disappear after a short period of simulation time, leading to a laminar flow solution.

Periodic boundary condition is applicable if the initial condition is perturbated with spatially correlated data. Having the periodic boundary condition saves the hassle of inflow data generation for each time step, and has been used in channel flows with success ([46], [92]). In simple flow conditions such as channel flow or duct flow simulations, the periodic boundary is applicable. Flows in complex geometries and bathymetry may not have the luxury of applying the periodic boundary condition. Flows with complex geometry and bathymetry may result in non-homogeneous turbulence, and thus inflow turbulent boundary condition is still required.

The inflow boundary condition chosen for this work was from the work of Klein *et al.* [77]. A digital filter was utilized to create two-point correlations between data points with a specified length and time scale. The digital filter is first defined as

$$u_m = \sum_{n=-N_f}^{N_f} b_n r_{m+n} \tag{2.14.1}$$

where,

r_m represent a series of data with zero mean, $\overline{r_m} = 0$, and unit variance, $\overline{r_m r_m} = 1$.

b_n are the filter coefficients,

N_f represents the spatial or temporal extent of the filter.

The autocorrelation function can then be written as

$$\frac{\overline{u_m u_{m+k}}}{\overline{u_m u_m}} = \frac{\sum_{j=-N_f+k}^{N_f} b_j b_{j+k}}{\sum_{j=-N_f}^{N_f} b_j^{\,2}} \tag{2.14.2}$$

This equation ties the filter coefficients and the autocorrelation function of u_m into one expression. The filter can also be extended for *3*-dimensional applications, simply by convolution of filters in the three spatial directions, resulting in

$$b_{jkl} = b_j \cdot b_k \cdot b_l \tag{2.14.3}$$

Klein *et al.* also showed another form of interpretation for equation (2.14.2) in an engineer's perspective. For normal application, the autocorrelation function is represented by the notation $R_{uu}(\boldsymbol{x},\boldsymbol{r})$ where r denotes a distance vector and $r=|\boldsymbol{r}|$, a magnitude distance or the radius of a sphere filter. Typically, r is chosen to be a multiple of the grid size. In the late stage of a homogeneous turbulence, the autocorrelation function for a fixed time is written [77] as

$$\begin{aligned} R_{uu}(r,0,0) &= e^{\left(-\frac{\pi r^2}{4L^2}\right)} \\ L = L(t) &= \sqrt{2\pi(t-t_0)} \end{aligned} \tag{2.14.4}$$

The above function satisfy the basic properties of $R_{uu}(0)=1$, and $\lim_{r\to\infty} R_{uu}(r) = 0$ and the length scale can be prescribed. The simplified autocorrelation function in discretized form is shown as

$$\frac{\overline{u_m u_{m+k}}}{\overline{u_m u_m}} = R_{uu}(k\Delta x) = e^{\left(-\frac{\pi(k\Delta x)^2}{4(n\Delta x)^2}\right)} = e^{\left(-\frac{\pi k^2}{4n^2}\right)} \tag{2.14.5}$$

with the filter coefficients

$$b_k \approx \frac{\widetilde{b_k}}{\sqrt{\sum_{j=-N_f}^{N_f} \widetilde{b_j}^{\,2}}} \quad , \quad \widetilde{b_k} = e^{\left(-\frac{\pi k^2}{4n^2}\right)} \tag{2.14.6}$$

The generation of the inflow data thus uses the simplified equation (2.14.6) along with equation (2.14.3) to yield a two dimensional arrays of spatially correlated data for every time step. Using the Taylor hypothesis, the data is also autocorrelated temporally.

The procedure for the digital filter generation of turbulent inflow data is easy to code and robust for different types of applications, however may not be divergence free. To ensure that the divergence free criteria is met, changes are made.

In this study, only two components of the velocities fluctuations were prescribed while the fluctuations in the third component were obtained from the continuity equation. This ensured that divergence free condition was satisfied and the fluctuating velocities contained energy at both high and low wave numbers. All turbulent flow simulations utilized the inflow boundary fluctuations outlined above unless otherwise stated.

2.15 - *Time Step and Stability Criteria*

In numerical modeling, it is customary to ensure that a reasonable time step size be chosen to produce optimal and stable simulations. Choosing large time steps may lead to instability while excessively small time steps require longer run time. The numerical model used in this study is nonlinear. Therefore it was difficult to perform a thorough stability analysis.

As shown in section 2.7, the explicit predictor-corrector method was used for the viscous and advection terms. For that reason, the Courant-Friedrichs-Lewy (CFL) stability condition has to be satisfied. The CFL condition states that the sum of the Courant numbers has to be less than or equal to unity [2] in each cell. The CFL condition is defined as

$$\sum_{x_i=1}^{3} \left| \frac{u_{\max} \Delta t}{\Delta x_i} \right| \leq 1 \tag{2.15.1}$$

In addition to the CFL condition, the summation of the diffusion number has to be less than *0.5* in each cell. This criteria is sometimes known as the Von Neumann criteria and is expressed as

$$\sum_{x_i=1}^{3}\left|\frac{\nu_t \Delta t}{\Delta x_i^2}\right| < 0.5 \tag{2.15.2}$$

The minimum of the temporal step sizes from Equation (2.15.1) and equation (2.15.2) will govern. Since turbulent eddy viscosities are not known a-priori, therefore equation (2.15.1) will be used to estimate the time step size in advance.

Equation (2.15.2) can be used as a check if the chosen time step is appropriate. Violation of either of the conditions may cause instability and thus a smaller time step will be needed. These stability conditions are required but do not necessarily guarantee stability [14].

CHAPTER 3

BUBBLE TRANSPORT MECHANISM

Chapter 3 attempts to cover all aspect of the air bubble modeling. The purpose of this chapter is to introduce and include air bubbles into the previously discussed numerical model. The chapter begins with a general overview of the methodology used to incorporate forces generated by air bubbles to the governing equations. The proposed strategy to include the modeling of air bubble dynamics can be summarized into a two step process. The first step involves the modification of the LES set of momentum equations to account for the influence of the air bubbles towards the fluid. To accomplish this, a source term is added to the vertical momentum equation to include the forces due of the air bubbles. The second step is to slightly modify the scalar transport equation to include the effect of air bubble rise. These steps will tie the hydrodynamics of the fluid to that of the air bubbles. This chapter also provides the equations used for air entrainment quantification from an overfall plunging jet.

3.1 - *Equations for Incorporating Air Bubbles in the Flow Field*

The governing LES momentum transfer equation (equation (2.3.12)) for fluid flow in Cartesian coordinate, shown again here, is

$$\frac{\partial u_i}{\partial t}+\frac{\partial (u_i u_j)}{\partial x_j}=-\frac{1}{\rho}\frac{\partial p}{\partial x_i}+\frac{\mu}{\rho}\frac{\partial}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)+\frac{\partial \tau_{ij}}{\partial x_j}-\frac{\partial (gh)}{\partial x_i} \tag{3.1.1}$$

with the continuity equation

$$\frac{\partial u_k}{\partial x_k}=0 \tag{3.1.2}$$

In actuality, air bubbles and water mixture are two different fluids and is discontinuous in space. In a microscopic view of the problem, the dynamics of each fluid phase can be described by writing a set of governing equations for the different phases. However, that is not the purpose of this work. The application of this work is mainly on

modeling large scale problems and thus, a more practical approach is to view the problem in a macroscopic perspective. In doing so, air bubbles and water is viewed as a mixture and thus was treated as one continuum that coexists everywhere in space [130]. Under this assumption, air bubbles are massless and retain its size regardless of its surrounding pressure. In addition, under low concentration, the incompressibility assumption will still be valid. These assumptions allow the same set of momentum and continuity equations to be used to describe the dynamics of the fluid mixture.

In flows where air bubbles are present, the liquid phase is affected by the total net force that the air bubbles exert on each control volume (computational cell), throughout the whole domain. One simple way to account for these forces is to include them into the momentum equations. To do so, the net momentum transfer from the bubble to the fluid has to be modeled and will be lumped into the source terms in the momentum equations. The source terms due to the air bubbles can be related back to the movement due to the density difference between the air bubbles and the fluid.

To obtain the source terms, the collective forces of air bubbles of a known concentration were calculated for each cell. These forces were then normalized based on the unit mass of a cell. The normalization produced the source terms required by momentum equations. Upon determination of the source terms, the effects of the air bubbles could be accounted for as part of the hydrodynamics solution to the problem.

Revisiting the momentum equation, the only change that was needed to be made, was to include the source terms due to the net force of the air bubble exerted onto the fluid flow field. Adding a new source term produced

$$\frac{\partial u_i}{\partial t}+\frac{\partial (u_i u_j)}{\partial x_j}=-\frac{1}{\rho}\frac{\partial p}{\partial x_i}+\frac{\mu}{\rho}\frac{\partial}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)+\frac{\partial \tau_{ij}}{\partial x_j}-\frac{\partial (gh)}{\partial x_i}+S_i \tag{3.1.3}$$

where, S_i is the net momentum exerted by the air bubbles on the fluid system. This procedure does not require rigorous arithmetic effort and can easily be implemented into most existing Navier-Stokes solver. The momentum contributions of the air bubbles accounted as the source term, S_i, in the LES momentum equations will be presented in the next few sections.

3.2 - *Acting Forces on Single Bubble*

The analysis can be started by looking at different forces acting on a single air bubble in a control volume. The free body diagram for a single air bubble in is shown in Figure 3-1.

The acting forces on a single air bubble consist of the buoyant force, drag force and weight of the air bubble, and are written as

$$F_{net} = F_b + F_d + W_t \tag{3.2.1}$$

where,

F_{net} is the net force acting on the bubble,

F_b is the buoyancy force,

F_d Drag force, and

W_t weight of a single air bubble.

The buoyant, drag and weight are forces which directly influence the flow domain in the present of air bubbles. The buoyancy force can be defined as the weight of liquid being replaced by the rising bubble. Thus,

$$F_b = \forall_g \rho_l g \tag{3.2.2}$$

The buoyancy force is the main driving force for the vertical movement of air bubbles. In this context, buoyancy force is a function of the volume of air bubble, the density of liquid in the surrounding and gravity. The mass of an air bubble can be calculated simply by multiplying the volume by its density. Therefore, weight will be mass multiplied by the acceleration of gravity and can be written as

$$W_t = \rho_g \forall_g g \tag{3.2.3}$$

In a static liquid, the buoyant force will be balanced with the drag force and the weight of the bubble. To simplify the analysis, the weight of the air bubble is neglected. Therefore, the two remaining force components are the buoyant force, being balanced by the drag force. Since the drag force is directly related to the forces on the surrounding fluid, the drag force will be modeled. Simplifying, F_b will be equal to -F_d.

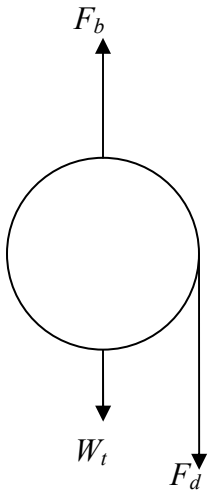

Figure 3-1: Forces on a single air bubble.

Assuming liquid is at static, the drag force on a body can be expressed as

$$F_d = -\frac{1}{2} C_d \rho_l V^2 A \tag{3.2.4}$$

where

C_d is the drag coefficient of the air bubble,

ρ_l is the density of the surrounding fluid,

V is defined as the velocity of the air bubble, and

A is the maximum cross sectional area of the air bubble that is shearing the fluid.

To differentiate between the gas and the liquid phase, subscript *l* will be used for liquid phase while subscript *g* will be used for gas phase in this chapter. Equation (3.2.4) can be rewritten to

$$F_d = -\frac{1}{2} C_d \rho_l V_g^{\,2} A_g \tag{3.2.5}$$

Considering a non-static fluid, then the drag force is directly related to the relative velocity of the bubble and the fluid. Therefore, the velocity, which is written in equation (3.2.4), is expressed in terms of relative velocity. By definition, relative velocity is the difference of the velocity of an object with its background velocity. Substituting this definition in equation (3.2.4) gives

$$F_d = -\frac{1}{2} C_d \rho_l (V_g - V_l)^2 A_g \tag{3.2.6}$$

or simply,

$$F_d = -\frac{1}{2} C_d \rho_l \left|V_r\right| V_r A_g \tag{3.2.7}$$

where $V_r = V_g - V_l$

Equation (3.2.7) is consistent with literature [71].

To further simplify the expression, it was assumed that the velocity of the air bubble achieved terminal velocity in a very short time span and thus, $V_g \approx V_\infty + V_l$. Here, the subscript ∞ represents the condition when the terminal state was achieved. The assumption that the air bubble quickly reaches its terminal velocity is reasonable.

Literature has shown that small air bubbles may reach its terminal velocity as fast as the first *200* milliseconds of its rise [106]. Therefore, substituting $V_g \approx V_\infty + V_l$ into equation (3.2.6) produces

$$F_d = -\frac{1}{2} C_d \rho_l (\left[V_g + V_l \right] - V_l)^2 A_g \tag{3.2.8}$$

Now, since $V_r \approx V_\infty$ then $C_d \approx C_\infty$. This simplification will give

$$F_d = -\frac{1}{2} C_\infty \rho_l \left| V_\infty \right| V_\infty A_g \tag{3.2.9}$$

The drag force in equation (3.2.9) is now independent of the fluid velocity. If V_∞ and C_∞ are known, then the drag force on a single bubble is easily determined. The relationship between the terminal velocity and the drag coefficient is discussed in the next section. Before proceeding to the next section, it is important to realize that the drag force given in equation (3.2.9) is in relation to the bubble. An equal and opposite force will be used as the force being added to the surrounding fluid.

3.3 - *Terminal Velocity*

The terminal velocity of air bubbles in water was calculated based on the following equations given by Bozzano and Dente [13]. To find the terminal velocity, V_∞, of a bubble with diameter D_g, the following equation can be used

$$V_\infty^2 = \frac{4}{3} \frac{g D_g}{C_d} \tag{3.3.1}$$

The drag coefficient of an air bubble is

$$C_d = f_f \left(\frac{a}{R_0} \right)^2 \tag{3.3.2}$$

and a is the radius of the air bubble in the horizontal plane and f_f is the friction factor. For a spherical bubble, a is equivalent to the radius of the bubble. The term $(a/R_o)^2$ is an adjustment factor for the deformation of air bubble. The generalized friction factor can be calculated by

$$f_f = \frac{48}{\mathrm{Re}} \frac{1+12Mo^{\frac{1}{3}}}{1+36Mo^{\frac{1}{3}}} + 0.9 \frac{Eo^{\frac{3}{2}}}{1.4\left(1+30Mo^{\frac{1}{6}}\right)+Eo^{\frac{3}{2}}} \tag{3.3.3}$$

where, *Eo* is the Eötvös number, *Mo* is the Morton number and is *Re* the Reynolds number. For clarity, the three dimensionless numbers are written as follows (with the subscript *l* being the liquid phase variables and subscript *g* for gas phase)

$$Eo = \frac{\left(\rho_l - \rho_g\right) g D_g^2}{\sigma}$$

$$Mo = \frac{g\mu_l^4}{\rho_l \sigma^3}$$

$$\mathrm{Re} = \frac{D_g V_\infty}{\nu_l}$$

where, σ is the surface tension of the liquid.

The Eötvös number, also known as the bond number is the ratio between the gravity force with the surface tension force [65]. The Morton number is the product of combination between three different dimensionless numbers, that is, Webber number, Froude number and Reynolds number [24]. The Morton number is also called the liquid parameter, since it is only a function of fluid properties and gravity constant. Together, these dimensionless numbers can be used to characterize the shape of an air bubble moving in fluid medium [35]. Hence, the friction factor in equation (3.3.3) can be thought of as the friction factor which takes into consideration the shape deformation of the air bubble. This trait was desirable for the current research to aid in providing a better overall model in estimating the drag created by each air bubble.

Although the ratio of a/R_o for a sphere bubble is equal to *1*, for more realistic simulation results deformation will or may occur. Hence the deformation was approximated by

$$\left(\frac{a}{R_0}\right)^2 \cong \frac{10\left(1+1.3Mo^{\frac{1}{6}}\right)+3.1Eo}{10\left(1+1.3Mo^{\frac{1}{6}}\right)+Eo} \tag{3.3.4}$$

An iterative procedure is required to solve for the terminal velocity. In this work, the Newton iteration was used and was easily incorporated into the code. Results generated from the equations for a wide range of bubble diameter sizes are presented in Figure 3-2. Bozzano and Dente suggested that the proposed expression is only valid if the Morton number is less than 10^{-8} [13]. The Morton number of 10^{-8} was well within the scope of this study as calculation showed that the Morton number for water at room temperature was several orders of magnitudes smaller than the applicability limit (approximately 10^{-11}).

Microscopic movements such as the oscillatory, zig-zag and helical motion have been extensively studied ([75], [108], [118]) and these secondary upward motion were neglected in this work due to their complexity. These secondary motions could not only change the terminal velocity of the air bubbles, additionally the drag coefficient could be affected. When simulating air bubbles in terms of concentration, which was the purpose of this work, it was difficult to track the secondary movements of each air bubble and include them as a bulk quantity. Other researchers are still working on the quantification of these relationships for a swarm of air bubble in a pure water-air system [107]. Modification could be done in the future when solid relationships are found for water-air system of swarm of air bubbles. For the scope of this work, only simple linear uprising motion was considered.

The results from various work were compared ([25], [28], [62], [91]), all showing good agreement. Hence equations (3.3.1) through (3.3.4) are used to quantify air bubble dynamics.

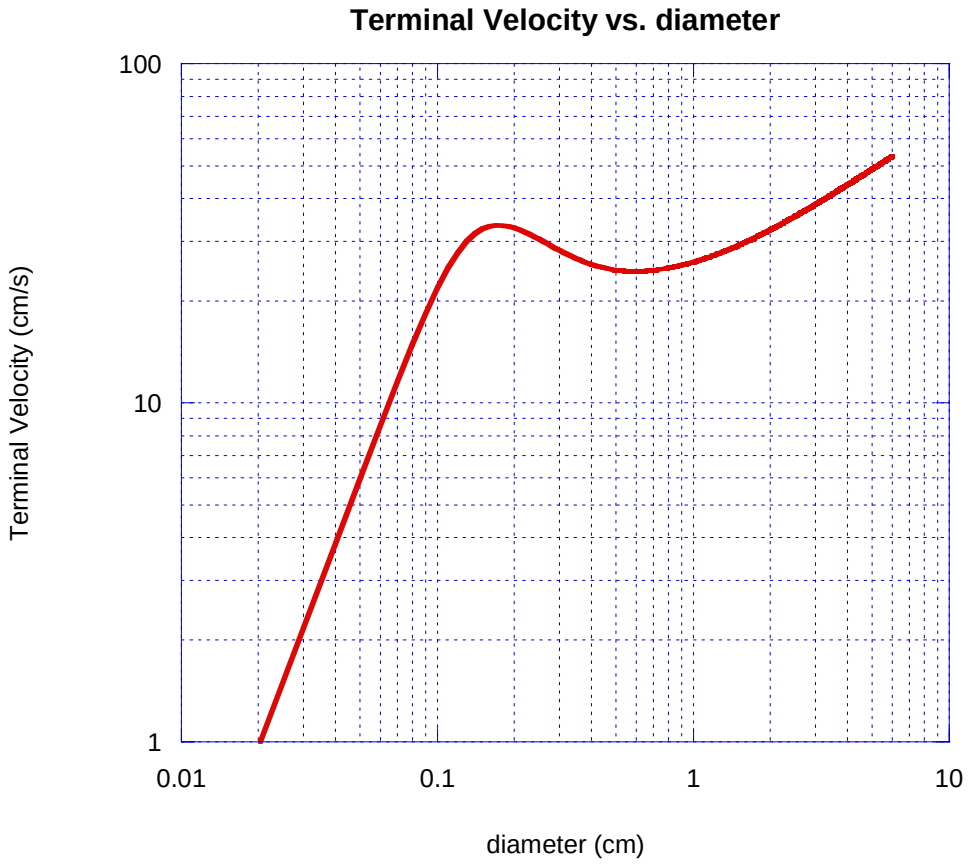

Figure 3-2: Terminal velocity obtained from theoretical equations (Bozzano and Dente [13]).

3.4 - *Link to LES Equations*

In the previous sections (sections 3.2 and 3.3), the terminal velocity and drag coefficient of a single air bubble was discussed. This information could be used to calculate the force exerted on the fluid by a single air bubble. The collective force of air bubbles in terms of concentration is now presented.

The collective forces of air bubbles on the fluid can be calculated by the summation of the forces of all air bubbles in a particular cell. Therefore, $F_{total} = -N\sum F_d$ will give the forces due to N number of bubbles in a control volume (cell). To incorporate the net force into the LES decomposed N-S equations, all the forces were expressed in terms of force per unit mass, namely

$$S_i = -\frac{N}{\rho_l \forall_l} F_d \tag{3.4.1}$$

where $\forall_l$ is the discretized volume of a cell (for a *3*-dimensional problem). N can be expressed in terms of concentration, ϕ by

$$N = \frac{\phi \forall_l}{\frac{4}{3}\pi\, r_d^{\,3}} = \frac{3\phi \forall_l}{4\pi\, r_d^{\,3}} \tag{3.4.2}$$

where, r_d is the radius of a spherical bubble. Substituting back into equation(3.4.1), the net source term S_i then becomes

$$S_i = -\frac{\left(\frac{3\phi \forall_l}{4\pi\, r_d^{\,3}}\right)}{\rho_l \forall_l} F_d = -\frac{3\phi \forall_l F_d}{4\pi\, r_d^{\,3} \rho_l \forall_l} \tag{3.4.3}$$

which can be simplified to

$$S_i = -\frac{3}{4}\frac{\phi}{\pi\, r_d^{\,3}}\frac{F_d}{\rho_l} \tag{3.4.4}$$

Now, S_i is the source term used in the LES decomposed equations for the vertical direction. Since air bubbles only have the tendency to rise due to buoyancy, there were no additional source terms to be considered in the horizontal momentum equations ($S_i=0$ for horizontal momentum equations).

3.5 - *Air Entrainment Quantification*

The amount of air entrainment had to be quantified based upon the information from the inflow plunging jet. Thus far, theoretical or analytical relationship of the air entrainment quantification is not well known. Only empirical relationships have been reported. Literature review in this area shows many relationships, mainly due to the lack of understanding in the precise mechanism that entrains air.

For nappes created from an overfall, Hadjerioua *et al.* [63] provided a semi-analytic expression

$$q_{air} = 3\times10^{-4} V_{jet}^{\ 2.1} t_h^{\ 0.6} \tag{3.5.1}$$

Where V_{jet} is the impinging velocity in meters per seconds and t_h is the nappe thickness in millimeters. However, this equation is not a general equation which is applicable for all conditions. Equation (3.5.1) is only applicable for nappes less than *60 mm*. For nappes greater than this limit, the equation for a vertical rectangular plunging jet is needed.

Mason [86] provided an equation that relates the volumetric concentration of air, intended for sheet or rectangular jets, and was by far the most relevant equation for this study. This equation provides a dimensionless relationship, given as follows

$$\phi_0 = 0.13\left(1-\frac{V_e}{V_{jet}}\right)\left(\frac{H_0}{t_h}\right)^{0.446} \tag{3.5.2}$$

where V_e is a constant of *1.1 m/s*, V_{jet} is the jet impinging velocity, H_0 is the drop height and t_h is the thickness of the jet. The application of this equation has been well documented and cited on several occasions ([18], [24], [86]). For valid application of this equation, the impinging velocity must be being between *1.5 m/s* and *9 m/s* [24]. Since equations (3.5.2) and (3.5.1) were well accepted in studies of similar plunging jet scenario, these equation were included as part of the air entrainment model.

Newer equations can be found, such as that provided in Melo [90]. Melo suggested a set of more updated and comprehensive equations to quantify the air entrainment for a plunge pool for a partially and fully developed jet, and was obtained from Bohrer and Abt [10] and Bohrer *et al.*[11]. However, the air concentrations given by Melo's equations

was significantly different from equation (3.5.2) and furthermore, its limitations were not clearly stated. Therefore, Melo's equations were not used for this study.

The task of quantifying the air entrainment is heavily dependant of empirical equations and this work merely utilized these relationships to provide the necessary boundary conditions for the plunging jet section. Once the air bubble concentration was provided, the burden of modeling the migration and movements of the air bubbles was placed on the numerical model itself.

3.6 - *Tracking Air Concentration*

The goal of this research was not to track every single air bubble, but to track bubbles based on concentration in every discretized finite volume. This was done by utilizing the scalar transport equation.

The scalar transport equation is similar to the transient continuity equation with a source term. Since the air bubbles are treated as scalar mass particles, the transport equation can be used effectively to track the concentration of the air bubbles in each cell. To begin a simulation, air bubble concentrations must be provided at the boundary of the plunging jet. Once the concentrations were provided at the boundary, the transport and migration of the air bubbles could be tracked by using the scalar transport equation

$$\frac{\partial \phi}{\partial t} + \frac{\partial \phi u_i}{\partial x_j} + D_T \frac{\partial}{\partial x_j}\left(\frac{\partial \phi}{\partial x_j} + \frac{\partial \phi}{\partial x_i} \right) = SS_{ti} \tag{3.6.1}$$

Here, tensor notation is used in the transport equation, and SS_{ti} are the source and sink terms for the transport equation respectively and ϕ is denotes the concentration of air bubbles.

Concentration can be expressed in several different ways. One way to express concentration is the volume fraction of gas to liquid and was used in this work. For an arbitrary scalar, no modification is needed to the equation shown above. However, gas bubbles have the tendency to move to the free surface and therefore, the motion of the air bubble is not only dependant on the flow field.

A simple adjustment was made for the bubble dynamics. Since the terms in the scalar transport equation basically means the flux change in control volume due to its respective direction, the uprising velocity of the air bubbles should also be included. Thus, the effective upward velocity becomes the velocity of the liquid plus the terminal velocity of the air bubbles. In the Cartesian coordinate system the effective upward velocity is

$$w_{sc} = w + V_{\infty} \tag{3.6.2}$$

In generalized coordinate, the effective upward velocity is

$$u_{\zeta sc} = u_{\zeta} + \zeta_z V_{\infty} \tag{3.6.3}$$

where the subscript *sc* is the modification of the vertical velocity for the scalar transport equation. For clarity, the scalar transport equation in Cartesian coordinate, for the vertical direction can be rewritten as

$$\frac{\partial \phi}{\partial t} + \frac{\partial \phi u}{\partial z} + \frac{\partial \phi v}{\partial z} + \frac{\partial \phi \left(w + V_{\infty} \right)}{\partial z} + D_T \frac{\partial}{\partial z}\left(\frac{\partial \phi}{\partial z} + \frac{\partial \phi}{\partial x} + \frac{\partial \phi}{\partial y} \right) = SS_{ti} \tag{3.6.4}$$

This modification ensured that the mass entering or leaving the control volume was consistent with upward velocity of the air bubbles and that of the fluid. No adjustments were needed for the velocities in the horizontal directions.

The source terms introduced the concentrations of air bubble into the control volume at the plunge section while the sink location for the air bubbles was at the free surface. At the free surface, the boundary condition was set as the Neumann boundary condition with zero flux. This allowed the air bubbles concentration to exit the domain through the free surface.

Lastly the source terms presented in equation (3.4.4) should not be associated with that of equation (3.6.4). The source term mentioned in equation (3.4.4) should only be coupled with the LES momentum equation while the source term in equation (3.6.4) is exclusive to the scalar transport equation.

CHAPTER 4

THE FREE OVERFALL IMPINGING JET

Following the treatment of the air bubble transport, the purpose of Chapter 4 is to provide information regarding the location, size, angle, and the velocities components at the impinging section. The mentioned boundary information is obtained through a sequence of steps which requires the upstream flow and drop height to be specified. The boundary conditions are then coupled with the air entrainment quantification to provide the complete set of boundary for an overfall jet. The last section in this chapter discusses the implementation details for boundary conditions of the overfall jet.

4.1 - *Free Overfall*

The free overfall is common in hydraulic structures. Free overfalls can be found in natural and manmade structures. Examples of the structures that produce the free overfall are spillways, waterfalls, weirs and other drop structures. In this work, the free overfall was essential in providing the necessary boundary conditions to describe the hydrodynamics phenomena in a pump station. The overfall profile was not part of the simulation. Therefore, the inlet boundary condition was directly specified at the impinging location to enable the simulation for the air entrainment, similar to Xu *et al.* [125]. The velocity impinging the water body was prescribed as inflow, while the amount of air entrainment due to the impinging water was obtained from empirical relationships described in the previous chapter, section 3.5.

The impinging location (location of the nappe) can be approximated by particle dynamics [101]. The equations for the upper and lower nappe are

$$-Z = a_{of} X^2 + b_{of} X - 1 \tag{4.1.1}$$

$$-Z = a_{of} X^2 + b_{of} X \tag{4.1.2}$$

respectively. The coordinates X and Z are in the horizontal streamwise and vertical directions, respectively, normalized by the brink depth. The coefficients a_{of} and b_{of} are approximated by

$$a_{of} = b_{of} = \frac{1}{2Fe^2} \tag{4.1.3}$$

where Fe is the Froude number at the brink before the drop, written as

$$Fe = \frac{Ue}{\sqrt{gD_{brink}}}$$

where,

Ue is the average velocity at the brink (flow/area), and

g is acceleration due to gravity

In order to make use of these equations, the upstream information had to be determined. Since the objective of this research was less concerned about the actual upstream dynamics of the impinging jet, an arbitrary condition for the upstream flow before the overfall was used and the depth at the brink can be approximated by [101]

$$D_{brink} = 0.715 D_{critical} \tag{4.1.4}$$

where, D_{brink} and $D_{critical}$ denote the brink depth and critical depth in the upstream channel, respectively. The critical depth can be calculated knowing the flowrate and dimension of the upstream channel and is given as [111]

$$D_{critical} = \left(\frac{Q^2}{b_w{}^2 g} \right)^{\frac{1}{3}} \tag{4.1.5}$$

where, Q is the flow rate of the upstream channel and b_w is the width of the upstream channel. The determination of the critical depth does not depend on the slope of the channel and therefore, the channel can be a horizontal channel.

To use equations (4.1.1) and (4.1.2) the drop height had to be specified. Once the drop height was specified, the horizontal distances of the impinging jet and the average velocity could be calculated.

Simulation of velocity impinging at different angles was also attempted, based on the flow condition and the drop height upstream. The lack of knowledge of the different velocities components within the nappe impact section makes it difficult to prescribe good inflow boundary conditions. However, with the impact angle known, the magnitude

and direction of the different velocities components can be approximated, and will be shown in section 4.2. In this analysis, the velocity in the lateral direction was assumed to be negligible.

4.2 - *Impinging Angle and Velocities*

One of the more important aspects for the plunging jet boundary conditions were the impinging velocities components. The upstream flow can be used to provide the location and size of the nappe, but from the hydrodynamics point of view, the needed boundary conditions are the velocities, broken down to its respective components at the entrance section. Experimental studies by Chanson [21] provide a way to separate the different velocities component from the bulk flow impinging the pool. The angle of the falling nappe to the horizontal-axis is given as

$$\tan(\chi) = \sqrt{2}\sqrt{\frac{D_{brink}}{D_{critical}}}\sqrt{\frac{h_{fall} + \frac{D_{brink}}{2}}{D_{critical}}} \tag{4.2.1}$$

where,

χ is the impinging angle,

D_{brink} is the brink depth,

$D_{critical}$ is the upstream critical depth, and

h_{fall} is the overfall drop height.

Thus, knowing the resultant velocity, the respective velocity components are determined by the angle given in equation (4.2.1). The velocities at the boundary are

$$\begin{aligned} u &= V_{jet}\left(\cos(\chi)\right) \\ w &= V_{jet}\left(\sin(\chi)\right) \end{aligned} \tag{4.2.2}$$

Chamani and Beirami [19], showed a similar relation of the velocity *u,* for a *2*-dimensional channel with a drop into a pool.

4.3 - *Overfall Jet Boundary Condition*

This section explains the logistics of the inflow boundary conditions at the plunging jet impact section. The plunging jet itself was not being modeled, therefore the jet impact section needed to be specified as a boundary condition. To characterize the jet impact section, several important components of the jet had to be determined. First, the jet dimension and location was determined. Then, the impact velocities were prescribed. Following that, the free surface boundary condition at the impact section was rewritten to account for the additional flux introduced by the plunging jet. At the impact section, the air entrained concentration will be estimated using an empirical relationship.

The location of the plunging jet was determined using equation (4.1.1) and (4.1.2). From equation (4.1.1) and (4.1.2) the lower and upper nappe location can be determined and the difference of the lower and upper nappe at the impact section gives the jet thickness. The impact angle was obtained from equation (4.2.1). The impact angle will be used to separate the entrance velocity into the different velocities components. Therefore, velocity at the entering section only consisted of two different velocity components, as suggested in equation (4.2.2). The velocities can then be used as the boundary conditions for the momentum equations. In scenarios where the depth changes at the impact location of the free overfall, the new averaged depth was used to calculate the jet location at the beginning of each time level, and the adjustment can be made to recalculate the location and angle of the jet, as well as the air bubble concentration level.

The free surface boundary condition was also altered to accommodate problems involving plunging jets. The simplest mode to account for the added flux by the jet can be written as

$$\frac{\partial h}{\partial t}+\frac{\partial}{\partial x}\left(\int_{-d}^{h} u dz\right)+\frac{\partial}{\partial y}\left(\int_{-d}^{h} v dz\right)=V_{jet} \tag{4.3.1}$$

This condition is only true for the plunging section. The variable V_{jet} is the average impinging velocity from the overfall. The average velocity will give a consistent mass flux across the jet section. Similar to the previous section (section 2.8), discretizing equation (4.3.1) gives

$$a_1 h_{j-1,k-1}^{n+1} + a_2 h_{j-1,k}^{n+1} + a_3 h_{j-1,k+1}^{n+1} +$$
$$a_4 h_{j,k-1}^{n+1} + a_5 h_{j,k}^{n+1} + a_6 h_{j,k+1}^{n+1} + a_7 h_{j+1,k-1}^{n+1} + \quad (4.3.2)$$
$$a_8 h_{j+1,k}^{n+1} + a_9 h_{j+1,k+1}^{n+1} = RHS + \Delta t(V_{jet})$$

Equation (4.3.2) was used as the boundary condition for the free surface equation at the jet impact section.

As explained in section 2.9, the dynamic pressure boundary condition at the surface was specified to be zero. In the case of a plunging jet, this condition is not true. The baseline assumption used was that the fluid is continuous from the jet to the plunge pool and thus specifying a zero pressure at the impact section was contradicting that assumption. The Neumann boundary condition was instead used for the pressures at the plunging section of the pool with the dynamic pressure flux equal to zero.

According to literature, air bubbles are generated at the boundary around the jet due to the free surface instabilities and the result of air water mixing that developed at the intersection ([21], [24]). Therefore, the air bubble concentration should be appropriately placed at the surrounding cells (of outside rim) of the intersection between the plunging jet and receiving body of water. As mentioned earlier in Chapter 2, scalar variables are located at the cell centers while the velocities are located in cell faces. Thus, the appropriate placement of the bubble concentration would be at the cells within the plunging jet edges, as shown in Figure 4-1. The migration of the air bubble was then modeled using the momentum and scalar transport equation.

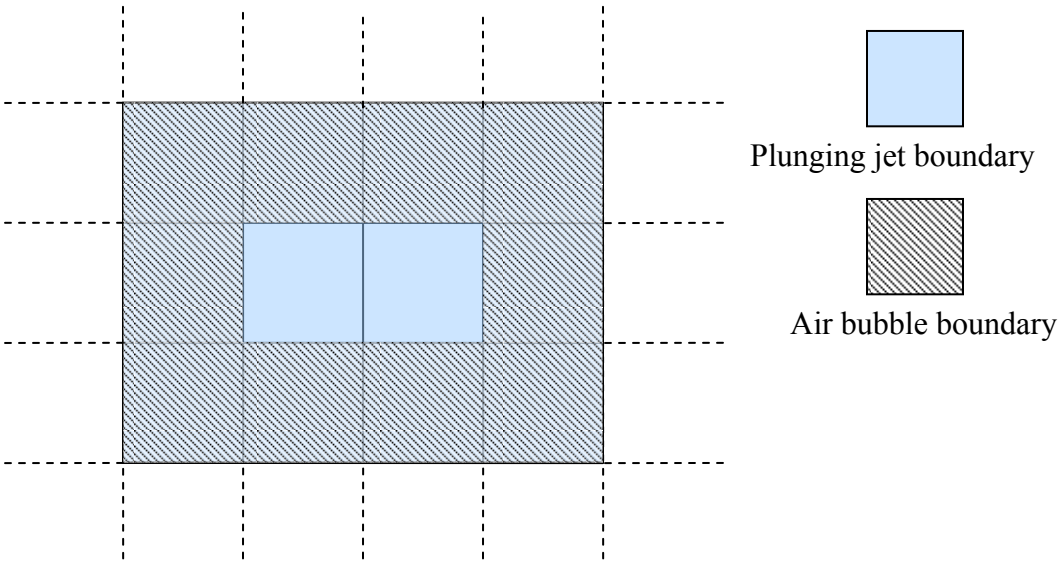

Figure 4-1: Placement of air bubble concentrations at plunging jet entrance boundary.

CHAPTER 5

RESULTS: VERIFICATIONS AND SIMULATIONS

A significant part of computer aided studies is the ability to reproduce analytic solution, experimental data or numerical results from literature. In pursue of this work, considerable effort was placed in the validation of the numerical code. This chapter deals mainly with several selected model verifications and the results of the final product. In this chapter, comparisons were conducted to show the validity of every aspect of the numerical code. The verifications will include laminar and turbulent flow simulations, free surface and dynamic pressure tests and the predictability of the bubble transport model. The chapter concludes with a simulation of a plunging jet with the presence of air bubbles.

5.1 - *Laminar Flow with Analytic Solution*

The model was first tested with laminar flow cases. Laminar flows are good case studies because of the availability of analytic solutions. Examples of such cases are the laminar Couette and Poiseuille flow ([55], [96], [102], [111]). Couette flow is basically flow between two parallel plates with one plate being stationary and the other plate moving at a certain speed. The Poiseuille flow has the same setup as the Couette flow, except that both plates are stationary. The analytic solution for the Poiseuille flow is

$$u = \frac{1}{\nu\rho}\frac{\partial \overline{P}}{\partial x}\frac{z}{2}\left(z - h_d\right) \tag{5.1.1}$$

where $z = h_d$ is the elevation at the top plate while $z=0$ is the elevation of the bottom plate. The others variables are self explanatory. The analytical solution for Couette flow is

$$u(z) = \frac{zu_0}{h_d} \tag{5.1.2}$$

where u_0 is the speed of the moving top plate.

These analytical solutions for laminar Couette and Poiseuille flow can be found in fundamental fluid mechanics textbooks ([55], [96], [102], [111]) and are well known. Analytic solutions are exact solutions derived based on theory and therefore will be good benchmark test cases to check the validity of the main code.

5.2 - *Laminar Poiseuille Flow Simulation*

The first test carried out is the simulation of flow between parallel plates, also known as the Poiseuille flow. The Poiseuille flow is a pressure driven flow, where the velocity profile depends on the pressure gradient. Inflow and outflow sections were set as fixed inflow and free flow boundary condition. There are basically two ways to perform this simulation. The first approach is to use a specified velocity profile and calculate the pressure gradient across the channel. The second approach is to provide a pressure gradient and let the flow evolve to a steady state velocity profile. The former approach was chosen, and the rationale for this choice explained shortly.

Since the analytic solution is well documented for this type of flow, the solution obtained from the simulation can be easily verified. Although the analytical solution is for a two dimensional case, simulations were done in a three dimensional environment. The free slip boundary was used for the side boundaries. The top and bottom boundary conditions were specified as no slip boundaries. The initial condition was provided with a velocity profile as in equation (5.1.1). The initial condition was provided for two reasons. The first reason was to avoid the shock scenario from a cold start which requires small temporal step to maintain computational stability. The second reason was to provide a background velocity to speed up the convergence to the steady state solution. Providing the analytical solution as the initial condition basically assumed that the flow is already developed, and therefore the correct pressure gradient would be calculated. In this case, a pressure gradient of $\frac{dp}{dx} = -1.5\,\frac{N}{m^3}$ was used to calculate the initial and inflow velocities. The size of the computational domain was $0.5\ m \times 0.25\ m \times 0.25\ m$ which was divided into $80 \times 40 \times 44$ uniform cells. The fluid viscosity and density were $\nu = 1 \times 10^{-4}\ m^2/s$

and $\rho = 1000\ kg/m^3$ respectively. The simulation was marched forward with a time step size of $\Delta t = 2.5 \times 10^{-3}$ until *10 sec*. Figure 5-1 shows the initial condition for the simulation along with the computational grids. Although not shown, the initial pressure was specified to be zero everywhere in the domain. The analytical velocity profile was maintained after *10 sec*. With the expected velocity profile maintained, as shown in Figure 5-2, the steady state pressure gradient was basically the same as that used to calculate the analytic solution. The next figure, Figure 5-3 shows the pressure drop along the x-axis. In the figure, the displayed pressure drop is actually $\frac{p}{\rho}$. Therefore, to get the correct pressure gradient, the values must be multiplied by a density term. At the outflow section, the pressure was found to be approximately $4.68 \times 10^{-6}\ N/\text{m}^2$ with $7.45 \times 10^{-4}\ N/m^2$ at the inflow. This translates to a $\frac{dp}{dx}$ of $-1.48\ N/m^3$, which closely matches the prescribed value of $-1.5\ N/m^3$, resulting in an error of about *1.3%*.

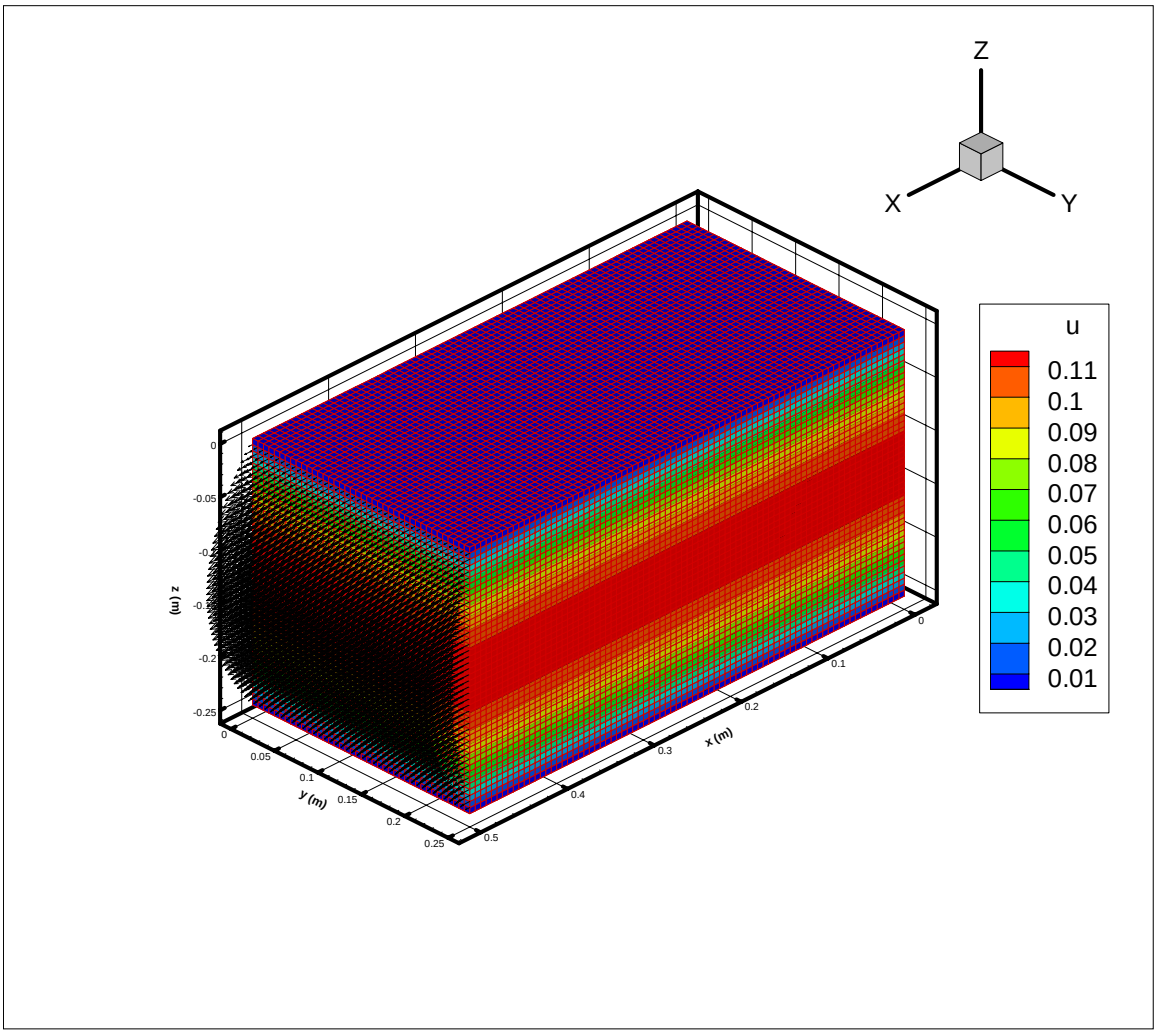

Figure 5-1: Poiseuille flow computational grid and initial condition.

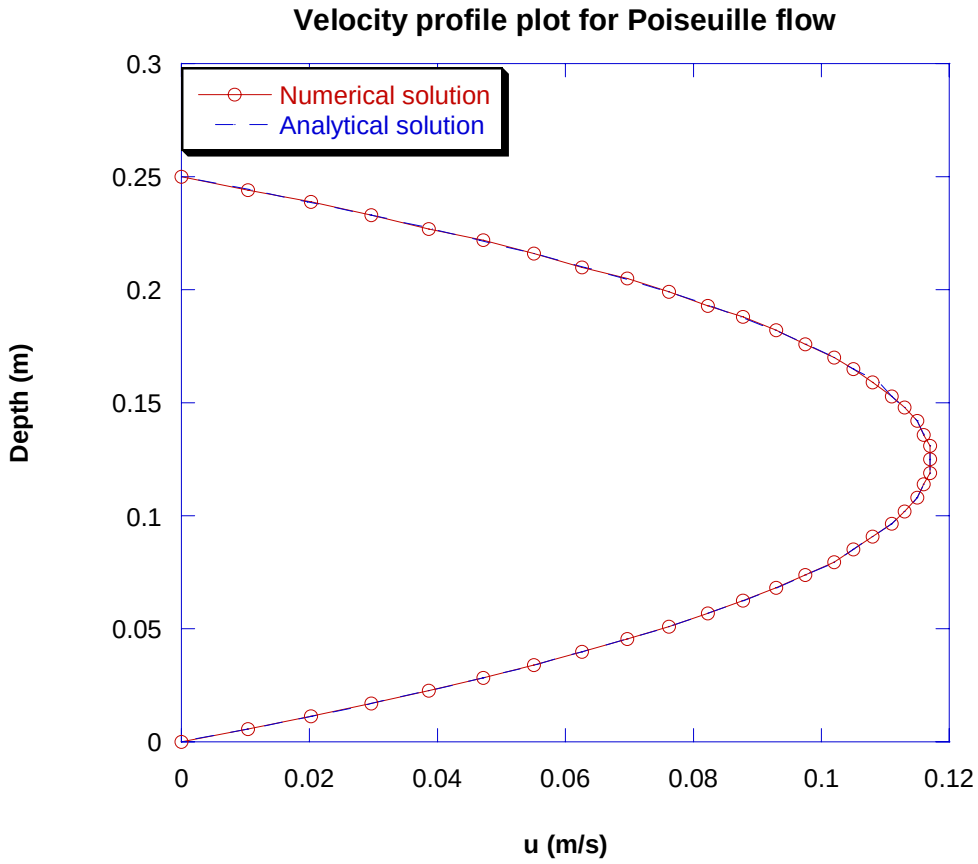

Figure 5-2: Two dimensional plot of velocity profile for Poiseuille flow.

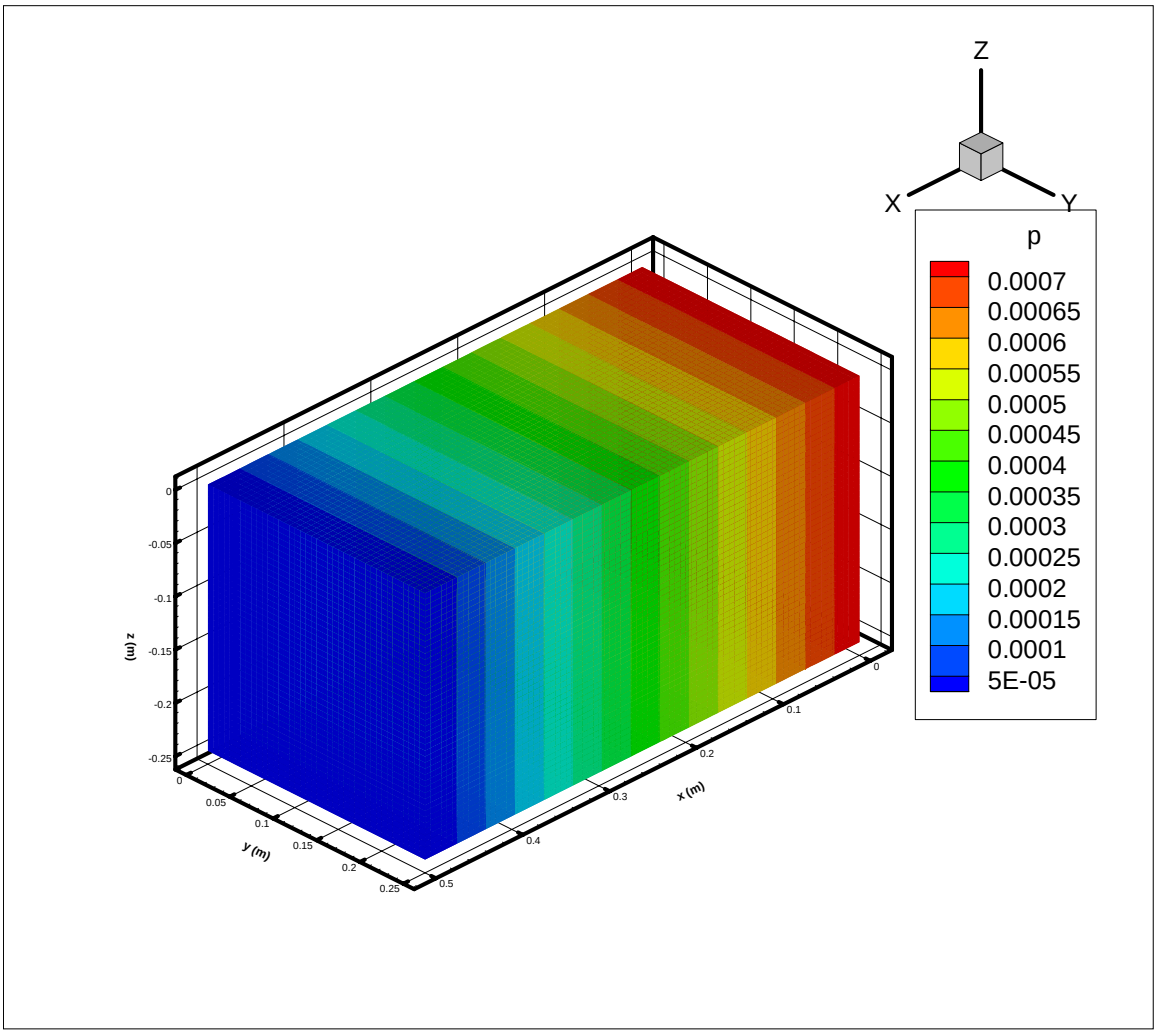

Figure 5-3: Pressure distribution of Poiseuille flow at steady state.

5.3 - *Laminar Couette Flow Simulation*

Another test for the laminar flow case is the Couette flow. Instead of a pressure driven flow, the flow is initiated and driven by the movement of the top lid. The conditions for the boundaries are set to be the same as the Poiseuille flow simulation. The only difference was that the top boundary was moving at a constant specified velocity of *0.1 m/s*. In this simulation, no pressure gradient was needed. The expected velocity profile would be linearly varying from the top plate to the bottom plate. The dimension of the domain was also selected to be the same as the previous simulation. The domain was divided into $20 \times 10 \times 11$ computational cells.

The coarseness of the grid forced a smaller time step, $\Delta t = 0.01\ sec$ to be used. Due to the cold start, the velocity profile attains steady state at about *800 sec*. The code was run until $t = 1000\ sec$. The results from the numerical simulation at the steady state condition also agree well with the analytic solution. Figure 5-4 and Figure 5-5 shows the simulated results. In Figure 5-5, the velocity profile matched the analytical solution and therefore provided confidence to continue the validation of the code with other test cases.

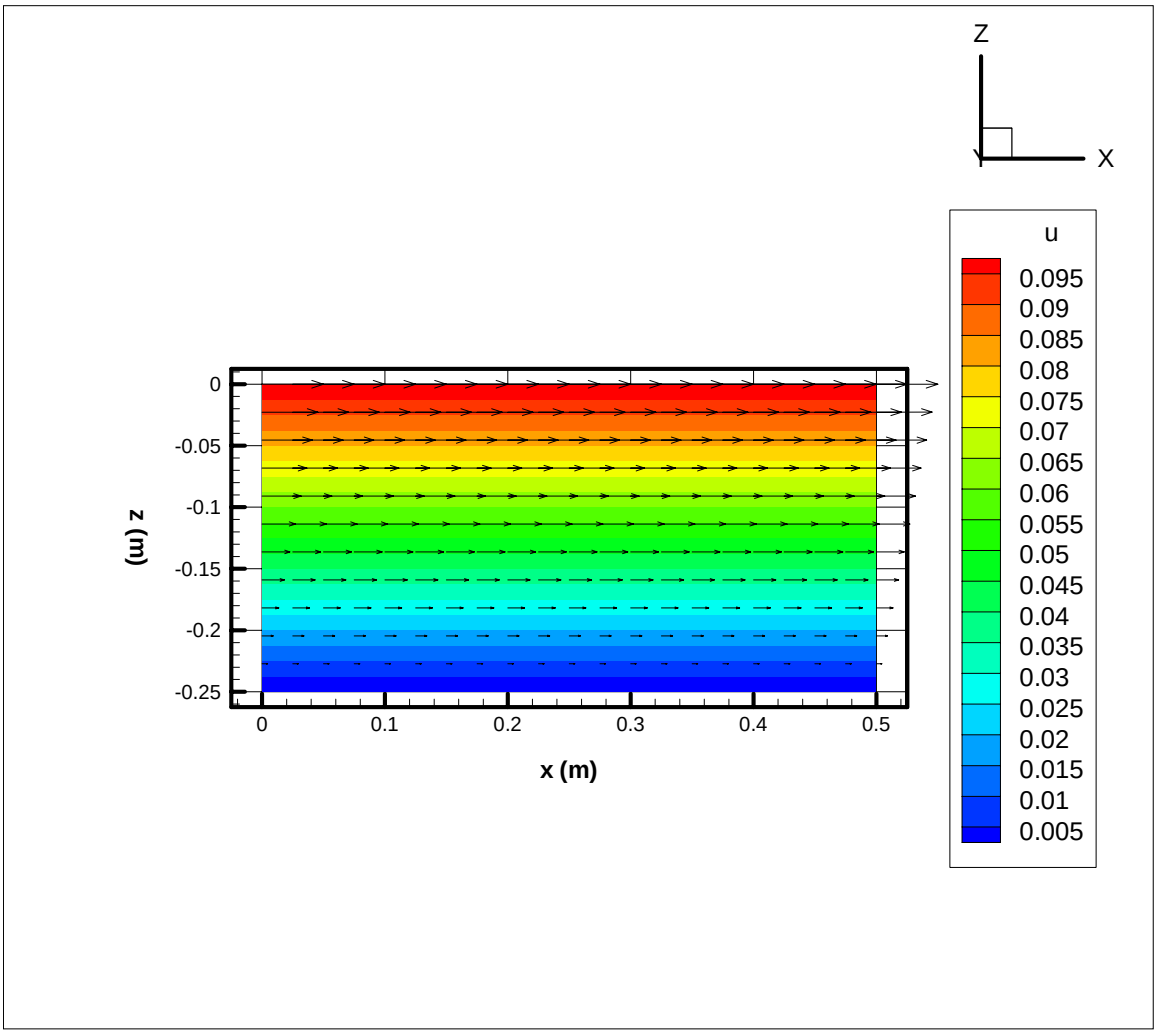

Figure 5-4: steady state solution for Couette flow.

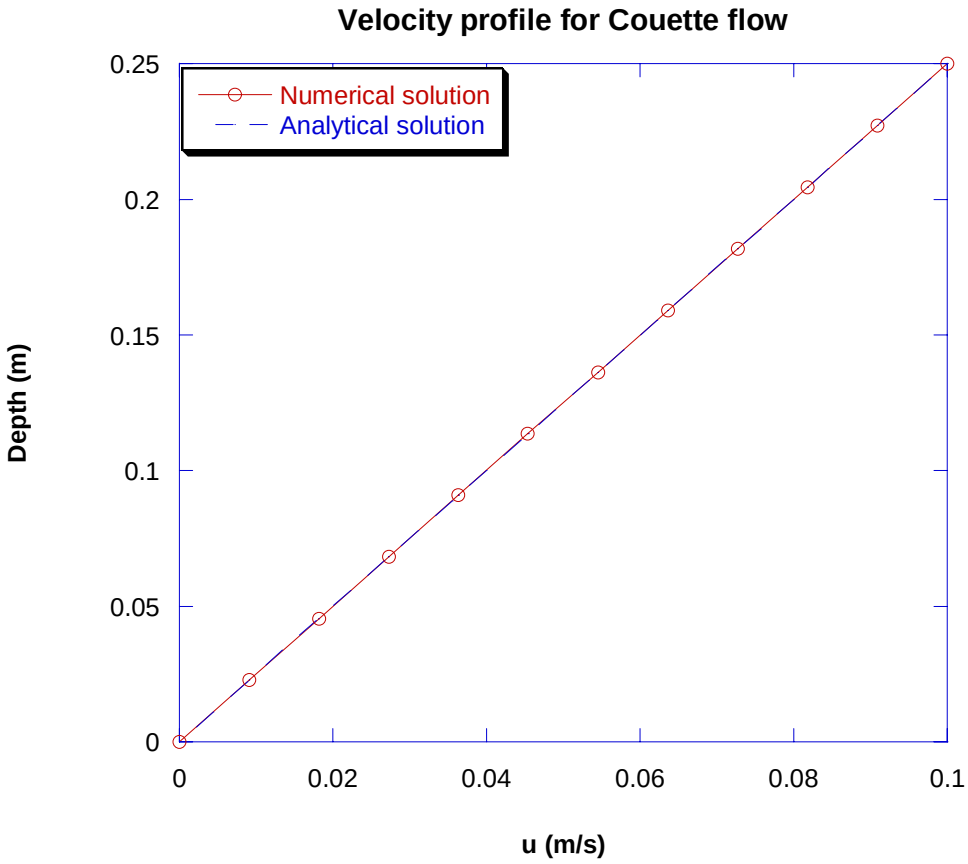

Figure 5-5: Steady state velocity at center of channel for Couette flow.

5.4 - *Laminar Open Channel Flow with Uniform Inflow*

The previous simulation were tests cases where the steady state solution is the same as that prescribed as the inflow. The next test case is a test case such that the velocity profile will be allowed to evolve to the desired analytic solution further away from a different inflow velocity profile. In this problem, the inflow was chosen to be a uniform profile and the initial condition was chosen to be the same as inflow for the same reasons mentioned previously in section 5.2. The simulation was marched until steady state was obtained and the expected velocity profile at the end of the channel should be the analytic solution. The reason for this test was to ensure that the results from the previous tests were not obtained mainly due to strong convection. Thus, if the profile evolves from a different inflow condition to the desired analytic solution, then the model is well validated for laminar flow conditions.

The molecular viscosity was chosen to be $\nu=1.137 \times 10^{-6}\ m^2/s$. The density of the fluid was set to $\rho=1000\ kg/m^3$. Uniform inflow and initial velocity profile was specified to be $9.1 \times 10^{-3}\ m/s$. Since this was an open channel flow problem, free surface and dynamic pressure were considered. The channel had a dimension of $4\ m \times 0.1\ m \times 0.05\ m$ and was discretized into $200 \times 10 \times 20$ cells in the x, y and z direction, respectively. Boundary conditions used for this run are free slip boundaries for the sides, constant uniform inflow and a fixed depth at the outflow. The bottom was specified as a no slip boundary. A temporal step size of $0.1\ sec$ was used, which was stable enough for the Crank Nicholson and the explicit schemes.

Figure 5-6 shows the velocity vectors at $t=100\ sec$, where the flow was still unsteady. After several hundred seconds, the flow achieved its steady state. The velocity vectors and dynamic pressure at steady state are shown in Figure 5-7 and Figure 5-8, respectively. For the open channel flow simulation the dynamic pressure for the fully developed region is small. At the inflow, transition from uniform profile to the parabolic velocity profile increases the dynamic pressure especially near the bottom of the channel.

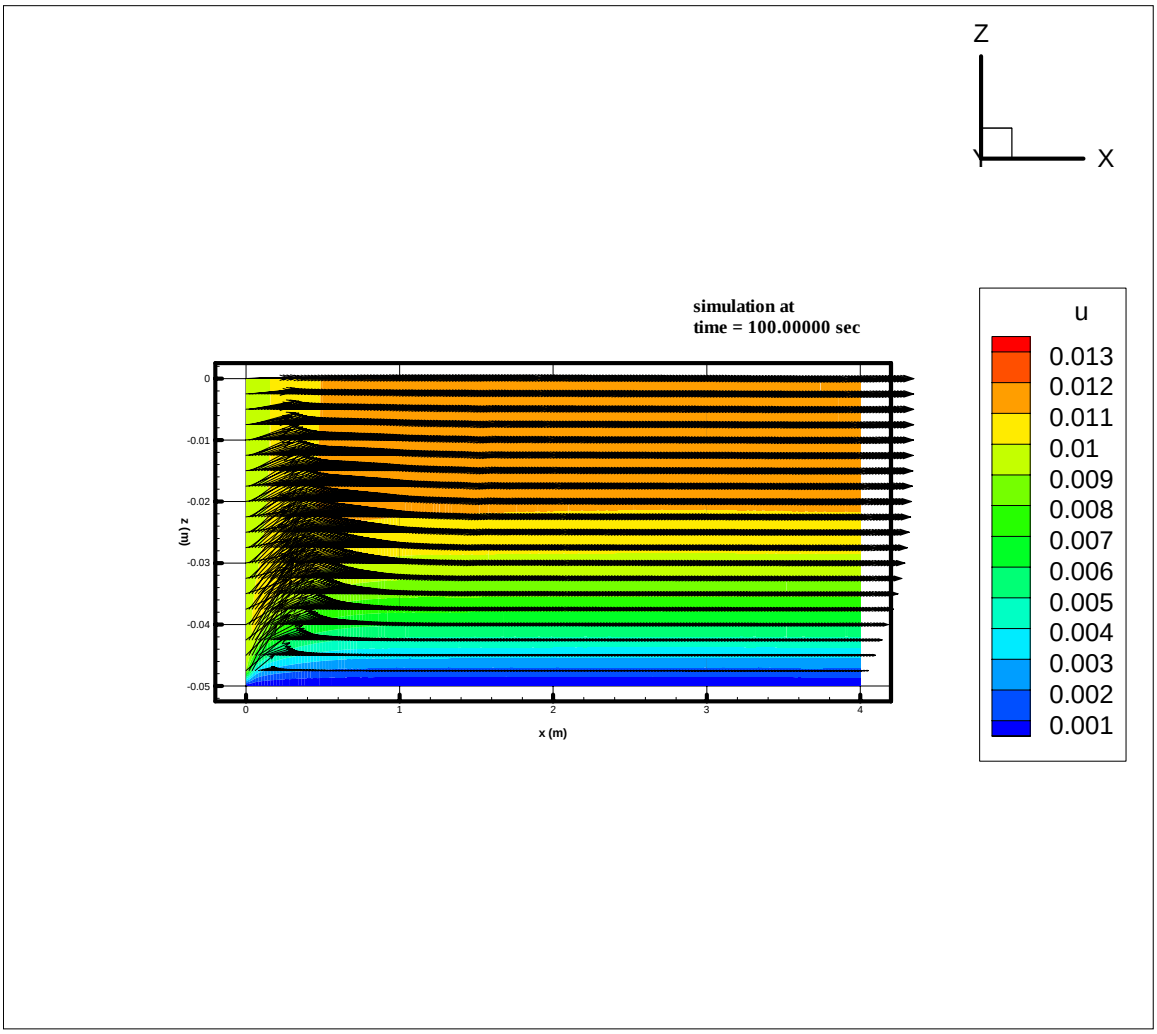

Figure 5-6: Velocity profile at *t=100 sec*.

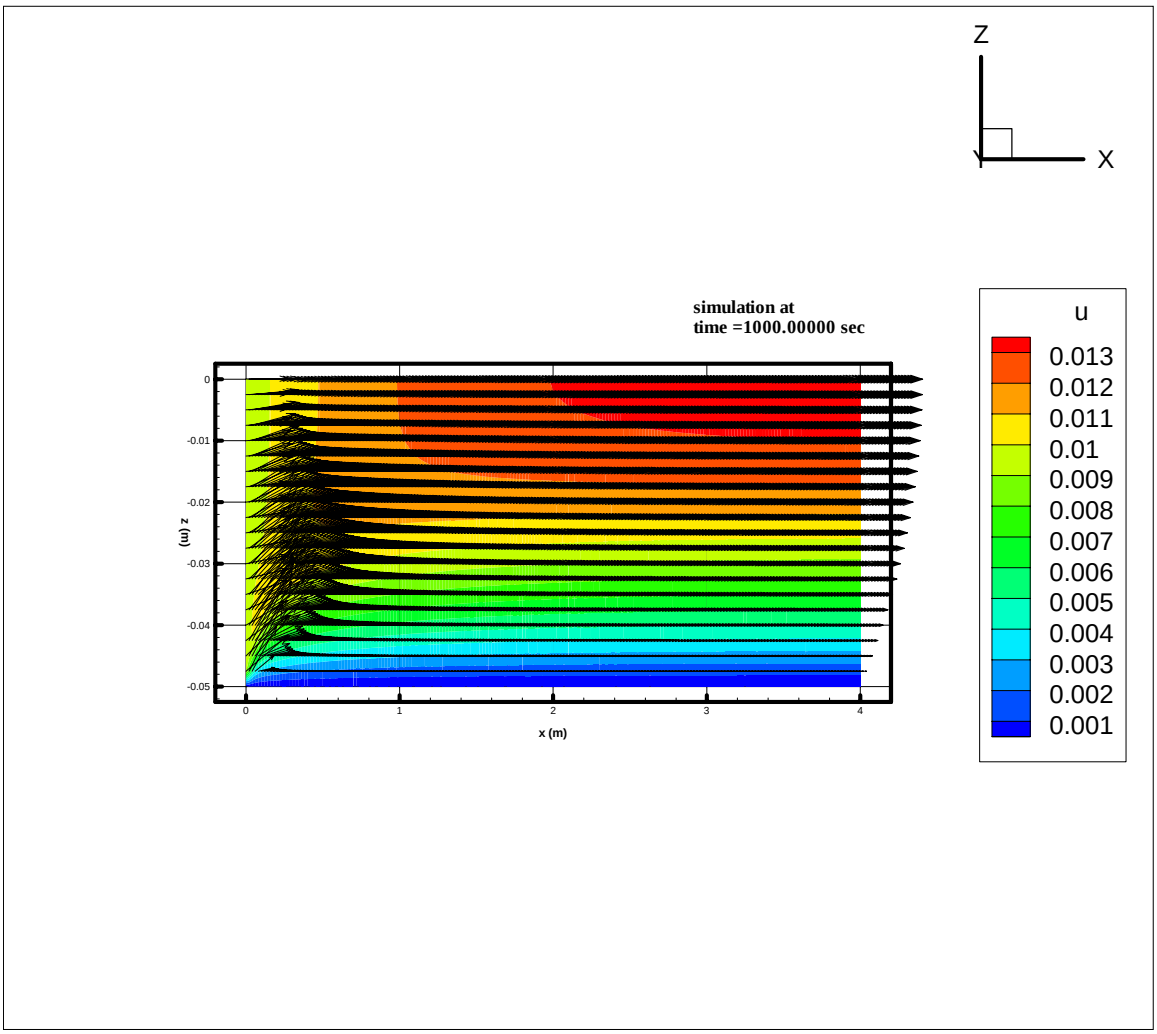

Figure 5-7: Velocity profile at steady state, *t=1000 sec.*

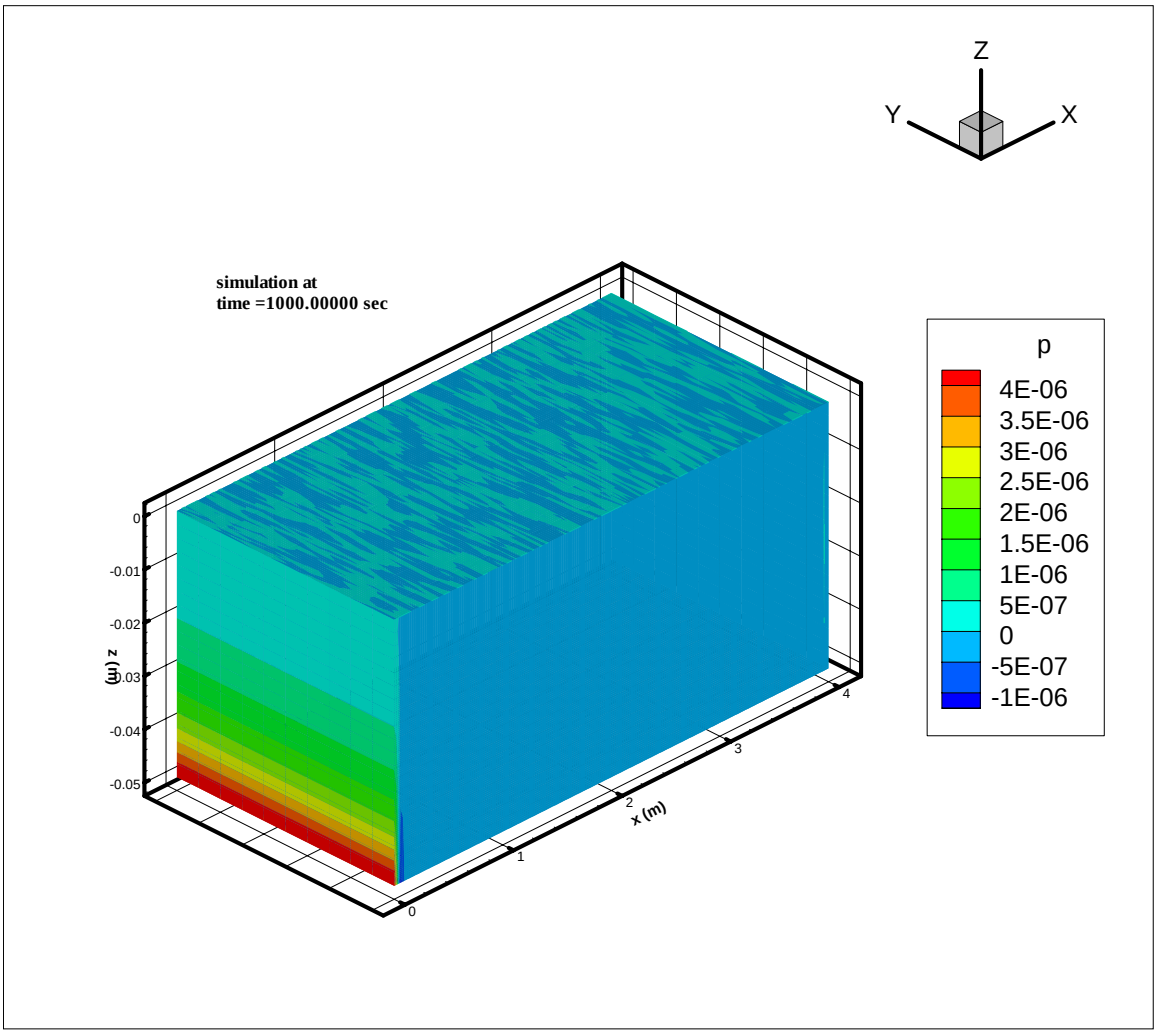

Figure 5-8: Dynamic pressure contours plot.

Figure 5-9 shows the analytical velocity profile and the numerical results obtained at steady state, at a distance of *x=80 m* from the inflow section. The side boundaries were free slip boundaries, therefore the flow is a *2*-dimensional flow with little variations across the width. The average velocity profile was calculated by averaging the velocities of the same depth laterally across the channel. The velocity profile matched the analytic profile everywhere except near the vicinity of the free surface. This behavior is mathematically justifiable with the way the boundary condition was applied. At the free surface, the Neumann boundary was used for the streamwise velocity, namely $\nu \frac{\partial u}{\partial z} = 0$, to define a zero shear for the free surface.

The mathematical implication of this type of boundary is that the velocity gradient across the boundary has to be zero. However, analytic solution for laminar open channel flow assumes that the velocity profile continues on for an infinite depth. The difference in the underlying boundary conditions is the cause of the discrepancy between the numerical and analytic solution.

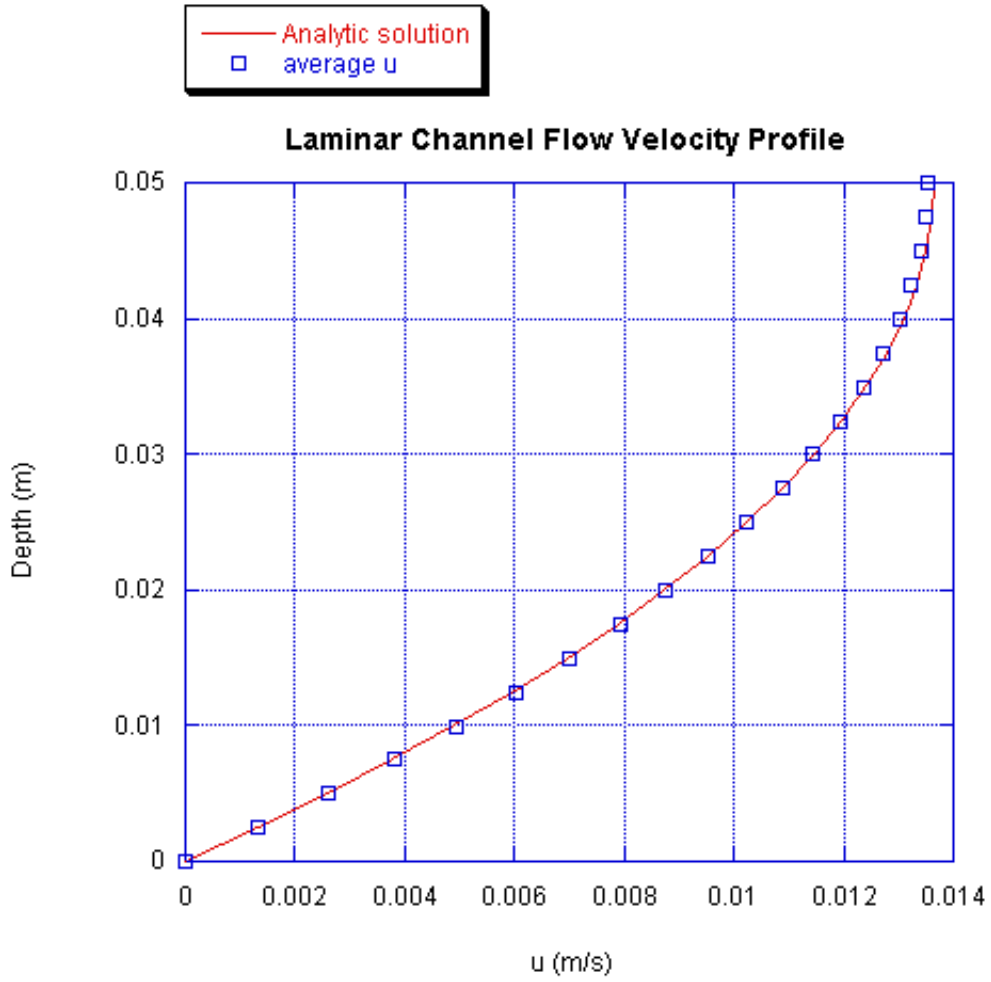

Figure 5-9: Laminar flow velocity profile at channel *x=80 m* from inflow.

5.5 - *Solitary Wave Run-up*

In order to test the free surface and hydrodynamics of the model, the solitary wave run-up test was performed. This simulation also tests the non-hydrostatic capability of the model. In this test, the two primary interests were the wave amplitude and the travel time, or speed, when the wave hits the wall. The travel time can be compared with the empirical equation in Ven Te Chow's book [32].

$$c = \sqrt{g\left(A_{wave} + D_0\right)} \tag{5.5.1}$$

where c is the celerity or speed of the wave, A_{wave} is the amplitude of the wave and D_0 is the still water depth. This equation gives the speed which can translate into the travel time of the wave. Chan and Street [20] later provided experimental data in which Tang *et al*.[112] used to compare his analytical solution with. Several other authors have made the same successful simulation as Chan and Street ([14], [70], [112]).

The expression for the analytical solution of the wave at different times can be found in Tang *et al*.'s[112] publication. The analytical expression is

$$\begin{aligned} h_{wave} &= A_0[\sec h^2\left(L+\delta_1\right) + \sec h^2\left(R+\delta_2\right) \\ &+ A_0\left(\frac{1}{2}\sec h^2 L \sec h^2 R - \frac{3}{4}\sec h^2 L \tanh^2 L - \frac{3}{4}\sec h^2 R \tanh^2 R\right)] \end{aligned} \tag{5.5.2}$$

where,

$$L = \frac{1}{2}\left(x - x_0 - \beta_T\right)\sqrt{3A_0}$$

$$R = \frac{1}{2}\left(x - 2x_d + x_0 + \beta_T\right)\sqrt{3A_0}$$

$$\delta_1 = \frac{1}{4}A_0\left(\tanh(R) - 1\right)$$

$$\delta_2 = \frac{1}{4}A_0\left(\tanh(L) + 1\right)$$

$$\beta_T = \left(1 + \frac{A_0}{2}\right)T$$

A_0 is defined as h_{wave}/ D_0, x_d is the location to the vertical wall and x_0 is the initial starting point of the wave (*i.e.* location of the wave crest at *t=0 sec*).

The wave was described with an initial surface elevation provided by the analytical expression. In this simulation, a *20m* long channel with a vertical downstream wall and still water depth of *1m* was used. The dimension of the lateral direction was set arbitrarily, chosen to be *1m* wide. The crest of the wave was initially at *10m* from the upstream boundary and the wave height was specified to be *0.1m*. Time step was chosen to be approximately $\Delta t=0.048$ seconds. In all simulations, the turbulence model was not used and an inviscid fluid was assumed. In the simulation, the domain was discretized into $62 \times 62 \times 30$ cells in the *x*, *y* and *z* direction, and θ of *0.5* (*i.e. Crank Nicholson*) was used. The side walls are free slip boundaries. A rigid wall boundary was used at the downstream end with a Neumann type boundary upstream. The no-slip boundary was used for the bottom boundary of the domain.

The dimensionless time scale used in the literature was stated as $T = t\sqrt{\frac{g}{D_0}}$, and was used as the reference time for comparison. The predicted run-up height, *R*, of the wave when it hits the wall is compared in Figure 5-10, which shows the averaged free surface profile of the wave for various times.

The simulation reproduced the results from the literature with good agreement. In the case of Tang *et al*. [112], the analytical solution given for the time of run up was $T = 9$ with an amplitude of *0.20151 m*. Figure 5-10 shows that the computed amplitude of the wave to be approximately *0.20446 m* at the same dimensionless time. The celerity also compared well with empirical equation found in Chow's book [32] (equation (5.5.1)). The equation yielded *3.28 m/s* while numerical results predicted approximately *3.26 m/s*.

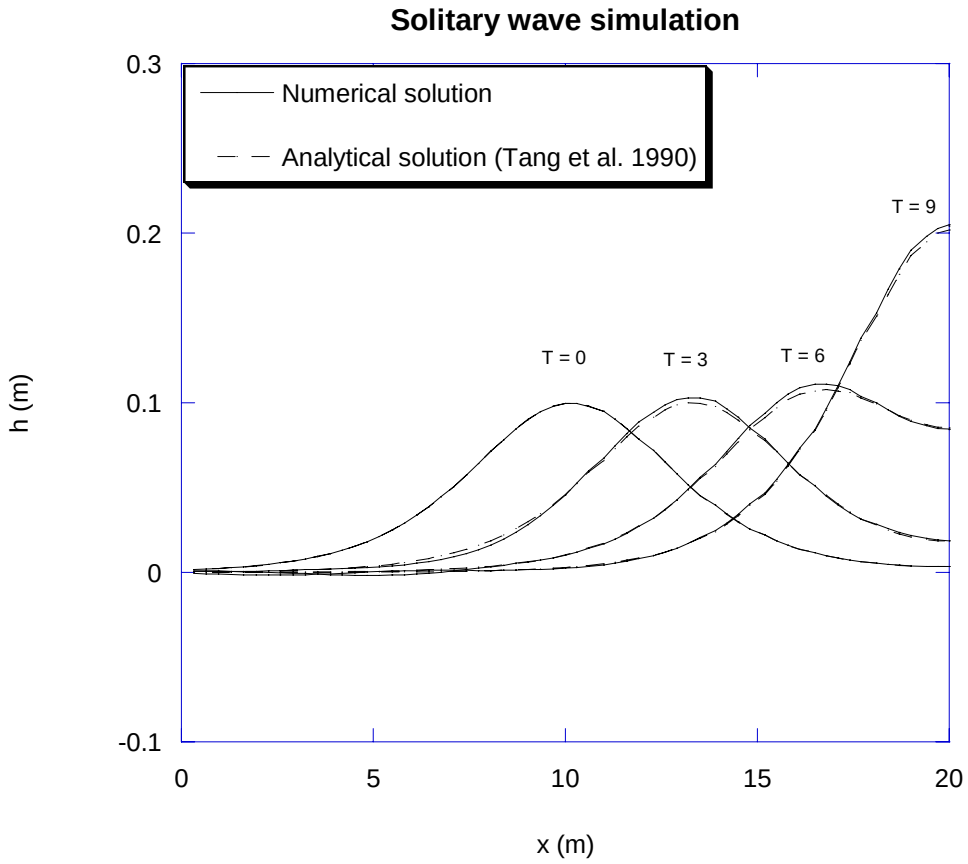

Figure 5-10: Surface wave propagation with dimensionless time scale.

5.6 - *Turbulent Channel Flow Simulation*

Turbulent channel flows have been studies extensively due to its simple geometry and flow nature. Experimental ([61], [126]) and numerical ([31], [87], [95], [131]) studies for this problem have been well documented. In turbulent channel flows, it is known that the velocity profile will behave according to the log law of the wall relation near the wall boundary [93]. This simulation serves two purposes. The first purpose was to validate the dynamic subgrid scale model while the second purpose was to test the code in a three dimensional setting. All previous simulations were merely two dimensional problems extended in a redundant third direction.

The code was tested with the same setup as Ding's [46] channel flow case study. The domain was made of two parallel plates with free slip side boundaries. Periodic boundary condition was employed in the streamwise direction. At the walls, no-slip boundary conditions were imposed. The dimensions of the domain can be described relative to the channel half width, δ. For historical reasons, the dimensionless quantities, scaled by the wall variables are occasionally used in this section. These dimensionless quantities are denoted by the superscript positive (+) sign. The wall variable for velocity is the shear velocity u_τ. previously defined in section 2.13. The wall variable for length is ν / u_τ and time is ν / u_τ^2. For clarity, generic form of these variables are shown as

$$u^+ = \frac{u}{u_\tau}, \quad y^+ = \frac{y u_\tau}{\nu}, \quad t^+ = \frac{t u_\tau^2}{\nu}$$

The streamwise and spanwise dimensions were $\pi\delta$ and $0.289\pi\delta$, approximately 570 times and 160 times the wall units ($\delta u_\tau/\nu$), respectively. These dimensions were the *minimal flow unit* used by Jimenez and Moin [74]. In this simulation, the domain was discretized with uniform cell size with $32 \times 32 \times 133$ cells. The discretization produced cell sizes of $\Delta x^+=17.6$, $\Delta y^+=5.11$ and $\Delta z^+=2.7$. For comparison, the shear Reynolds number, Re_τ , was chosen to be 180, which was used to calculate the mean pressure gradient.

The shear Reynolds number is defined as

$$\mathrm{Re}_\tau = \frac{u_\tau \delta}{\nu} \tag{5.6.1}$$

and the wall shear is defined as

$$\tau_w = u_\tau{}^2 \rho = \delta \left(\frac{\partial p}{\partial x_i} \right) \tag{5.6.2}$$

The streamwise pressure gradient could be explicitly calculated for verification purposes. The friction velocity, u_τ, from equation (5.6.1) was found to be 9.2124×10^{-1} *m/s* and the molecular viscosity used was a constant of 5.118×10^{-3} m^2/s.

The initial condition was provided with a two dimensional mean flow with a three dimensional disturbance, similar to Moin and Kim [92] and Ding [46]. The velocity field was added with fluctuations in the three different directions

$$u(x,y,z) = C_L \left(1 - x^8\right) + \varepsilon L_x \sin(\pi z) \cos(\frac{4\pi x}{L_x}) \sin(\frac{2\pi y}{L_y}) \tag{5.6.3}$$

$$v(x,y,z) = -\frac{1}{2} \varepsilon L_y \sin(\frac{4\pi x}{L_x}) \sin(\pi z) \cos(\frac{2\pi y}{L_y}) \tag{5.6.4}$$

$$w(x,y,z) = -\varepsilon (1 + \cos(\pi z)) \sin(\frac{4\pi x}{L_x}) \sin(\frac{2\pi y}{L_y}) \tag{5.6.5}$$

where C_L represent the mean velocity, $\varepsilon = 0.1 C_L$, L_x, L_y are lengths of the channel. The length scales were non-dimensionalized with the channel half width, thus resulting in the vertical (z-direction) coordinate bounded by ± 1 as suggested by Moin and Kim [92]. The bulk Reynolds number, *Re* was approximately *3126* and was calculated based on the mean velocity, C_L, and channel half width, δ. In this work, mean streamwise velocity profile was specified to be *16 m/s*. Table 5-1 summarizes the details of this simulation.

Table 5-1: Summary of the details used for channel flow simulation, $Re_\tau=180$.

Re	3126.23
ν	$5.117 \times 10^{-3}\ m^2/s$
ρ	$1000\ kg/m^3$
δ	$1\ m$
H	$2\ m$
U_{CL}	$16\ m/s$
dp/dx	$-8.4868 \times 10^{-1}\ N/m^3$
u_τ	$9.2124 \times 10^{-1}\ m/s$

To check for local mass conservation the fluctuation terms in equations (5.6.3)-(5.6.5) were differentiated and summed to ensure the divergence free condition was met before implementation. The dynamic subgrid scale turbulent model was used for this simulation. The time step was approximately *0.0008* time units, δ/u_τ, slightly larger than *0.0003* time units as suggested by Zhao and Voke [131]. The computation was carried out for about *12* time units, approximately *13 sec*.

In this simulation it was important that the results should reproduce streamwise velocity that follows the log law while maintaining the correct characteristics for turbulent eddy viscosity as shown in Vreman's work [121] and the profile of dynamic model coefficients that resembles the characteristic in Zhao and Voke's [131] and Ding's [46] result. In comparing the streamwise velocity profile, there are three distinct regions: the viscous sublayer ($y^+<5$), the buffer layer ($5<y^+<30$) and the inertial sublayer ($y^+>30$) [113]. In the viscous sublayer, the velocity profile must be linear and should follow that of the $y^+ = u^+$ curve. The velocity profile in the inertia sublayer must follow the log law relationship. In the buffer layer, the velocity profile should be a smooth transition that connects the viscous sublayer to the inertia sublayer velocity profile.

Streamwise velocity profile obtained from the simulation is plotted along with the log law of the wall formula. The plot was shown in Figure 5-11. The velocity profile near the solid boundaries matched the $y^+ = u^+$ curve while away from the boundary, the log law profile was recovered with only slight over prediction. A smooth transition was also observed within the buffer layer. The over prediction at the inertia layer could have been the cause of using a course grid resolution for this simulation, especially near the wall boundaries.

To ensure that mean dynamic model coefficient, *C*, obtained from the dynamic SGS model behave reasonably, numerical results from Zhao and Voke [131] was used as comparison. The plot in Figure 5-12 shows the comparison of the dynamic model coefficient obtained from this simulation with that of Zhao and Voke's data. The overall trend and magnitude of mean dynamic coefficient agrees reasonably with that of Zhao and Voke's data.

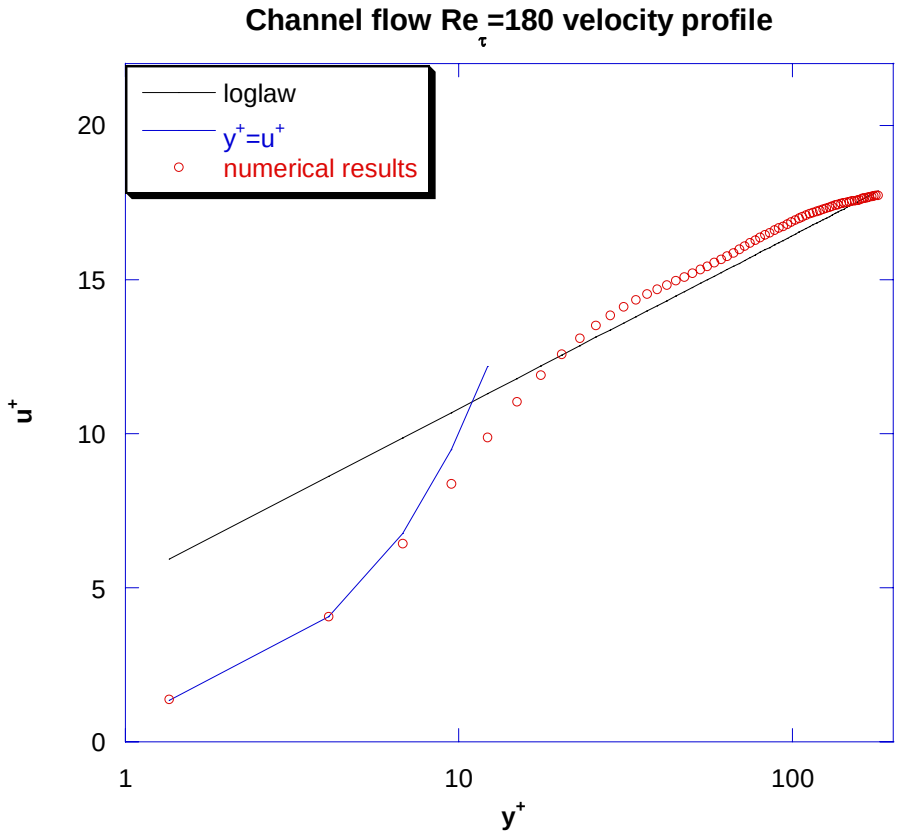

Figure 5-11: Mean velocity profile in wall coordinate for channel flow $Re_\tau = 180$.

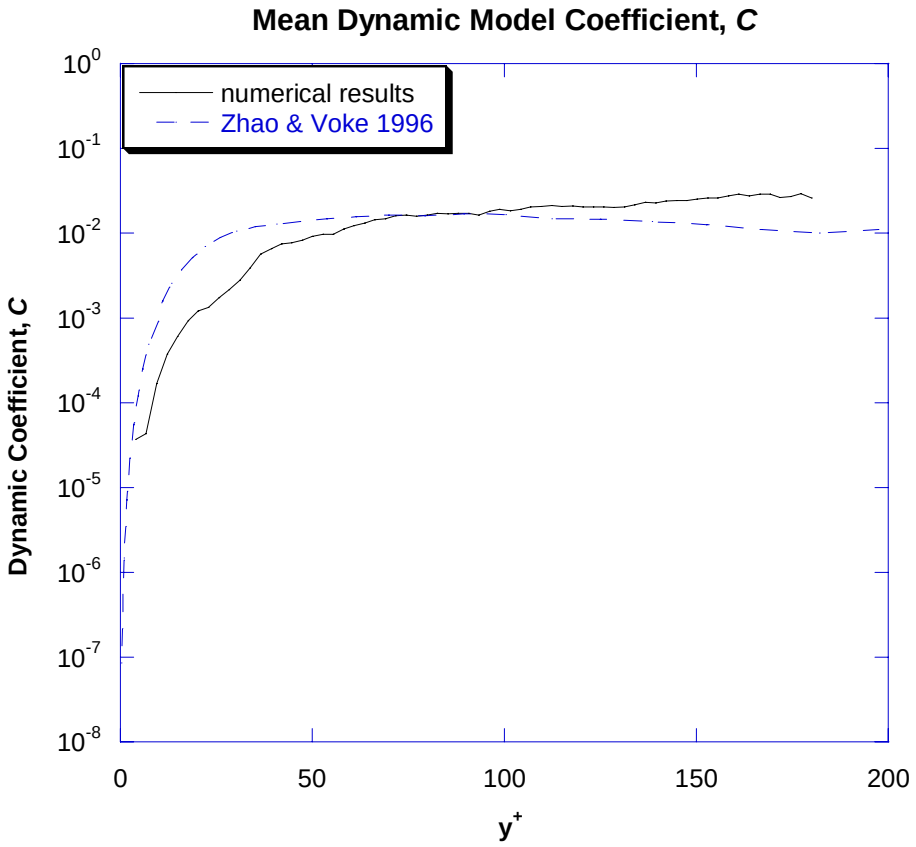

Figure 5-12: Mean dynamic model coefficient, C in wall coordinate for channel flow $Re_\tau = 180$.

As seen in Figure 5-12 the difference in the dynamic coefficients observed at $y+<80$ is about an order of magnitude. This difference is not uncommon for there are numerous factors which can cause different values of the dynamic coefficients when comparing numerical results from works. The more significant factors are the difference in numerical scheme and grid resolution, or combinations of both.

The numerical scheme used in Zhao and Voke's study, as well as, the vertical grid setup was different from this simulation. When comparing the dynamic coefficient for two different numerical schemes, differences are expected because of the unequal amount of numerical dissipation introduced by the different numerical schemes. Furthermore, when different grid sizes were used to simulate the same problem, the dynamic coefficient will be different. By definition, the dynamic model coefficient is a function of the filter width, also related to the grid sizes. The difference in grid size leads to different values of the dynamic model coefficients, but should produce eddy viscosities that correlates with the correct mean streamwise velocity profile.

More importantly, the numerical result produced the important characteristic of the dynamic subgrid scale model, which is the reduction of several orders of magnitude in the dynamic model coefficient as the wall is approached. This feature is an improvement from that of the traditional Smagorinsky model, and a clear advantage of the dynamic model when the grid resolution is adequate. The overall magnitude of the dynamic Smagorinsky coefficient agrees reasonably with that of Zhao and Voke's data.

Since the Dynamic Smagorinsky SGS model was used for this simulation, the eddy viscosities are observed to be a problem due to the sign changes from a positive eddy viscosity to a negative eddy viscosity in a rapid manner, at random locations, particularly when the simulation was first started. A simple treatment is to use a "clipping" method. This method, while particularly simple, has little theoretical basis other than providing stability for the computations. Ghosal *et al.* [58], and Ferziger [54], both are well aware of the disadvantages of the clipping method and suggested better treatment in their publications. Applying the clipping method to control the "negativity" of the eddy viscosity translate to a physical meaning that the backscattering process will not have a magnitude larger than the molecular viscosity itself. Backscattering is known to be physically possible [29] and backscattering can be advantageous in turbulent models.

However excessive backscattering causes the loss of stability and may lead to unphysical solution.

Other alternatives are available [80] to compute the eddy viscosity terms, however most have added computation expense. Preliminary investigation using the "clipping" method showed promising results for this particular simulation.

The focus of this work was not particularly on turbulence modeling. The author was merely implementing existing turbulence theory and models. Therefore, the mentioned turbulent SGS models with the underlined Boussinesq eddy viscosity assumption was used, and more sophisticated and improved turbulence modeling can be considered in the future. For the time being, result obtained from this run agrees reasonably well with published data.

5.7 - *Channel Flow with Air Bubbles*

One of the main components of this work was to trace the migration and transport of air bubbles. Little published work can be found with data that is usable to validate this part of the work. The only literature that has reasonable applicable value for this work is the publication by Zarrati [130]. Zarrati have performed some experimental work to obtain air bubble concentrations at different locations along an open channel. He also performed a *2*-dimensional numerical simulation of a channel flow with air bubbles. The experiment data was used for validation of his numerical model. His experimental data was also used for the validation of bubble transport model in this work.

In Zarrati's experiment, a channel with a downward slope of 14.5^{o} was used. The channel was *0.15 m* wide with a water depth of approximately *0.0225* meters. A sluice gate was located at the upstream end of the channel with an opening of *0.03* meters. Through direct email communication with the author, the channel wall roughness, k_s, was reported to be $1.25 \times 10^{-4}\ m$. At *1.5 m* downstream of the sluice gate, air was injected into the channel with an air diffuser which is located at the bottom of the channel. Data was collected from a reference plane downstream of the air diffuser, and every *0.1 m* apart until *0.6 m* downstream of the reference plane. The average velocity was reported to be *4.3 m/s*. In Zarrati's numerical simulation, the velocity profile at the inlet of the channel

was prescribed based on the power law, with the turbulent boundary layer thickness of approximately *10%* of the flow depth. The *k-ε* turbulent closure model was employed in his simulation. The size of air bubbles injected was not reported in the experiment data, however was mentioned in his numerical simulations to be of diameter greater than *2 mm* and less than *2 mm*, corresponding to terminal velocities of *19 cm/s* and *12 cm/s*, respectively.

This particular simulation was done using the classical Smagorinsky model with calibrated C_s value of *0.8*. The reason for this choice is that supercritical flow is a difficult problem to simulate with the dynamic subgrid scale model. The scale of the problem in addition to the high Reynolds number of the problem suggested the use of extremely huge number of grid points, which is currently not feasible. A single node processor would not permit the calculation to be carried out in a reasonable amount of time. Choosing the classical Smagorinsky model would not require such refined resolution, with the expense of using a crude turbulence model. The calibration involves matching the Smagorinsky parameter, C_s to the correct hydrodynamics.

For comparison, Zarrati's experimental data was used at the inflow section of the channel. Interpolation was done to obtain the values of air concentration for each cell centered node. Table 5-2 shows the data obtained from Zarrati's experiment as well as the interpolated values of air bubble concentration. In Table 5-2, "*Cell #*" column denotes the computational cell number starting from the bottom of the channel, "*zc*" represents the elevation at the center of a cell and "*conc*" represent the concentration at the initial section. The data without cell numbers are data sets from Zarrati's experiment while data with cell numbers are linearly interpolated air bubble concentrations. Interpolation was done as needed, assuming that the distribution of air concentration varies linearly between the two known corresponding experimental data. In Zarrati's computational work, *20* vertical grids were chosen, while *100* horizontal grids were used. The same Δz is chosen in this work.

Table 5-2: Air bubble concentration used in the verification simulation with values interpolated from Zarrati's experiment [130].

Cell #	*zc (cm)*	*Conc*
1	*0.056*	*7.916*
2	*0.169*	*8.647*
3	*0.281*	*9.378*
	0.300	*9.500*
4	*0.394*	*10.109*
	0.500	*10.800*
5	*0.506*	*10.888*
6	*0.619*	*11.750*
	0.700	*12.400*
7	*0.731*	*12.103*
8	*0.844*	*11.753*
	0.900	*11.500*
9	*0.956*	*10.178*

Cell #	*zc (cm)*	*Conc*
10	*1.069*	*7.645*
	1.100	*6.800*
11	*1.181*	*5.608*
12	*1.294*	*3.958*
	1.400	*2.400*
13	*1.406*	*2.678*
14	*1.519*	*2.065*
15	*1.631*	*1.747*
16	*1.744*	*1.429*
17	*1.856*	*1.112*
18	*1.969*	*0.794*
19	*2.081*	*0.477*
20	*2.194*	*0.159*

Since the experiment involved a sluice gate which causes transition, the upstream condition (sluice gate) was not included the domain. Instead the inlet boundary condition was set to have an average flow rate of $0.0145\ m^3/s$ and a fixed depth of $2.25\ cm$, the normal depth used by the author. The channel was $15\ cm$ wide and $60\ cm$ long, with a slope of 14.5^o. At the bottom of the channel, the quadratic drag wall function was used to avoid the near wall high velocity gradients. At the outlet of the channel, an open boundary for the flow and a Neumann boundary condition for the free surface were used. The domain size was $0.6\ m \times 0.15\ m \times 0.023\ m$. The domain was discretized into $100 \times 25 \times 20$ number of cells in the x, y and z direction, respectively, while the temporal step size was chosen to be $\Delta t = 1 \times 10^{-4}\ sec$. The temporal step satisfied the CFL stability condition. An a priori evaluation of the Von Neumann criteria (equation (2.15.2)) was not possible as mentioned earlier, in section 2.15. However, after several iterations the spatial and temporal step chosen produced stable runs and is used throughout the entire simulation.

The inflow velocity profile was specified to be a logarithmic profile obtained from the log law relationship (equation (2.13.1)) with $u_\tau = 0.564\ m/s$. Initial conditions follow the same inflow logarithmic velocity profile everywhere in the domain to avoid excessive shock associated with a cold start simulation.

Before the air bubbles were introduced, it was crucial that the hydrodynamics of the problem is simulated correctly. In some sense, the convection from the nonlinear terms would naturally have strong influence in the distribution of the air bubbles. Making sure that the velocity profile and depth agrees with open channel flow theory was a good check to ensure that the correct physics are captured. This was also used to ensure that the implementation of the bottom wall shear using the quadratic drag law relationship (equation (2.13.7)) behaves accordingly. The velocity profile can be compared to the log law profile while the depth can be compared to the average depth from manning's equation.

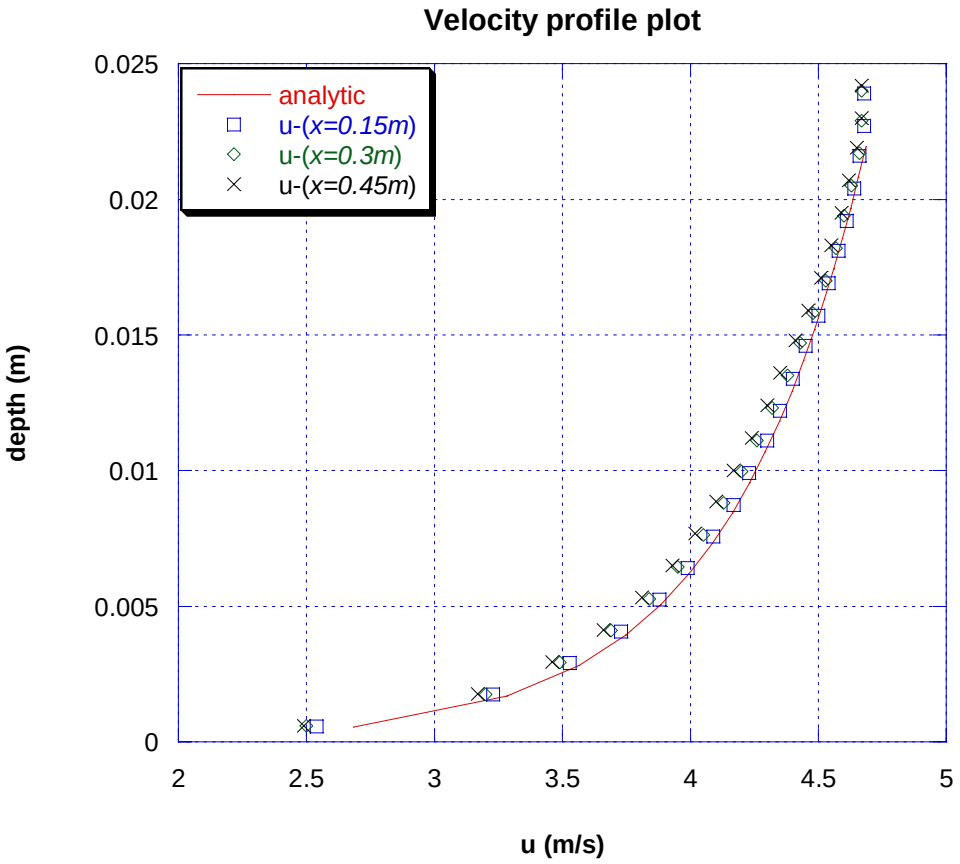

Figure 5-13: Streamwise velocity profile comparison with analytic solution.

The averaged velocity profile from the simulation is obtained and compared with the log law velocity profile in Figure 5-13. Here, *u-(x=0.15 m), u-(x=0.3 m)* and *u-(x=0.45 m)* are averaged velocity profiles obtained at *x=0.15 m, x=0.3 m* and *x=0.45 m* of the channel length measured, from the upstream boundary. Normal depth found using the manning's equation was approximately *2.3 cm*. The numerical result calculated a depth of approximately *2.36 cm*.

In his paper, Zarrati presented the experimental data along with a plot of the concentration contours obtained from his numerical model. Comparison can be made with both the available information. General trend of the air bubble concentration can be compared quantitatively, while a more detailed comparison can be done using the provided experimental data. From the experimental measurements the author mentioned that concentrations greater than *5%* were mainly of large air bubble, with diameter greater than *2 mm*. Concentration which are below *5%* shows most air bubbles with diameter less than *2 mm*. This observation proved to be useful information in the later part of this validation case study. The initial setup of the simulation is shown in Figure 5-14.

The simulation was started with the assumption that the flow in the domain is already at steady state and thus, the changes in free surface elevation will be small as reported in Zarrati's paper. A small wave was initially generated within the domain and soon passed the outflow boundary. The shock was caused by the changes to the flow condition, mainly due to the bottom boundary and the air bubble dynamics. Results were collected only after the shock had passed through the outflow boundary. The solution obtained from this simulation was compared to Zarrati's experimental data. Numerical results were taken at the center of the channel, along the *X-Z* plane (parallel to the flow direction). It is important to state that two simulations were carried out, the first with air bubble diameter of *0.75 mm* and the other with air bubble diameter of *0.72 mm*, which corresponds to the terminal velocities of approximately *13.17 cm/s* and *12.12 cm/s*, respectively.

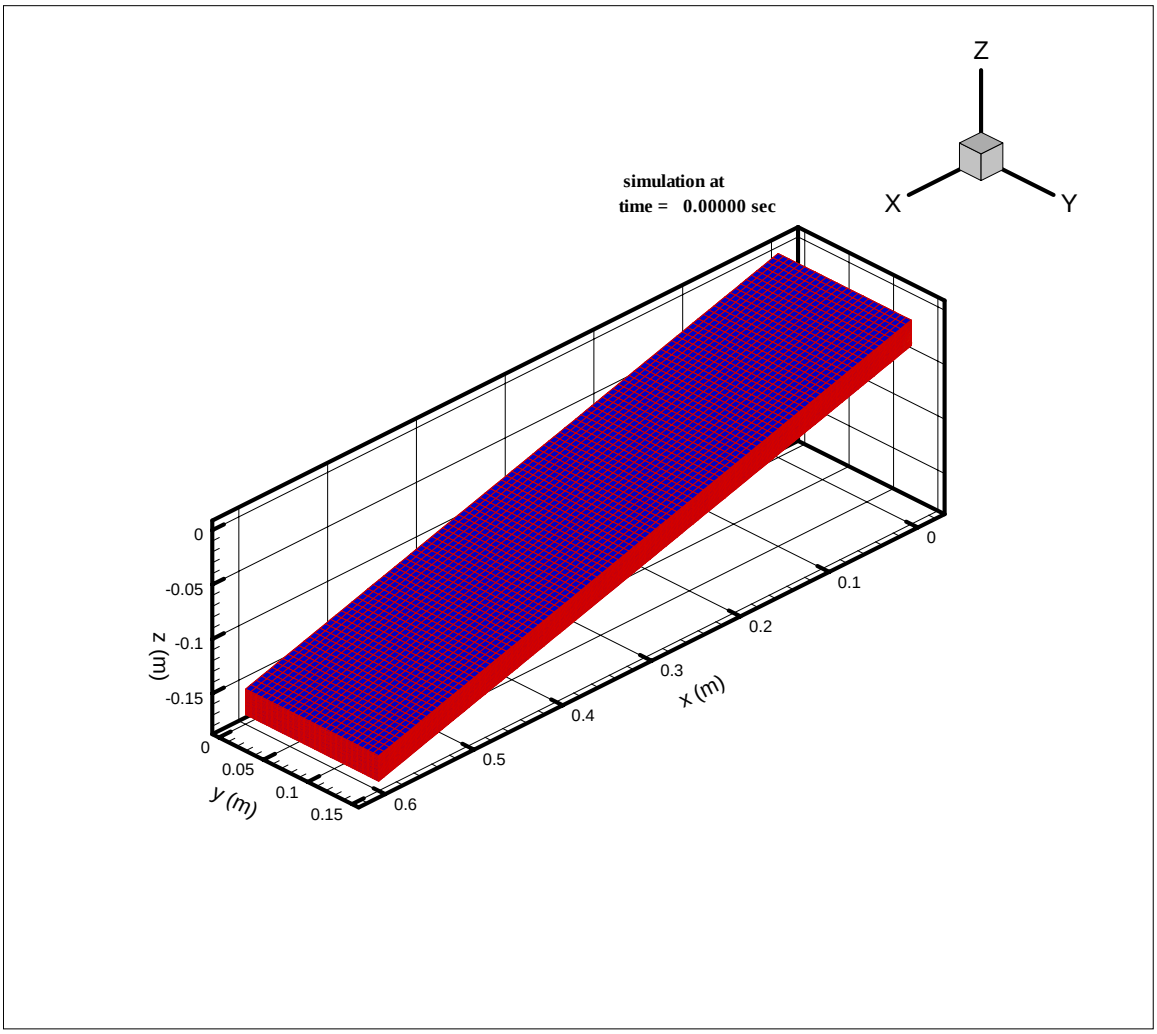

Figure 5-14 : Zarrati's simulation – domain discretization (*t=0 sec*).

The distribution of the air bubbles concentration was observed to have little change after several thousand time steps (approximately *2,500*). The primary interest and basis of comparison lies on the center of mass of the air bubble concentration. Since the bubble concentration profile does not vary much after approximately *2,500* time steps, further simulation was unnecessary. A typical concentration profile from the simulation is shown in Figure 5-15. Here the channel is plotted with respect to its depth at *t=0.25 sec* (*2,500* time steps).

Theoretically speaking, if there is no dispersion of the air bubbles, the air bubble concentrations will be an angled line from the inflow plane to the surface, producing banded lines for the different concentrations. Due to dispersion, the air bubble diffused both vertically and horizontally resulting in a more disseminated pattern, noticeably in higher concentration regions. Similar behavior is also expected in this particular simulation, given that the Froude number was high (greater than *1*), which encourages strong turbulent characteristics.

Quantitatively the steady state bubble concentration profiles had similar traits as that of the experimental data and Zarrati's *2*-dimensional simulation results. Figure 5-16 shows direct comparison of the experiment concentration with the two modeled concentration for air bubble diameter of *0.75 mm* and *0.72 mm*, at depths of *3 mm* and *15 mm* away from the channel bed.

Tabular form of the data is presented for detailed evaluation of the numerical model. Table 5-3 and Table 5-4 show the qualitative comparison of Zarrati's experimental data and the numerical results for *0.75 mm* diameter air bubbles, while Table 5-5 shows the percent difference between the two. The formula used to calculate the % differences is

$$\% \, Diff = \frac{Conc_M - Conc_E}{Conc_E} \tag{5.7.1}$$

In equation (5.7.1), $Conc_M$ is the modeled concentration while $Conc_E$ is experimentally determined concentration. In evaluating the performance of the air bubble transport model, Table 5-5 can be used only as a preliminary comparison due to the fact that only a single diameter was simulated.

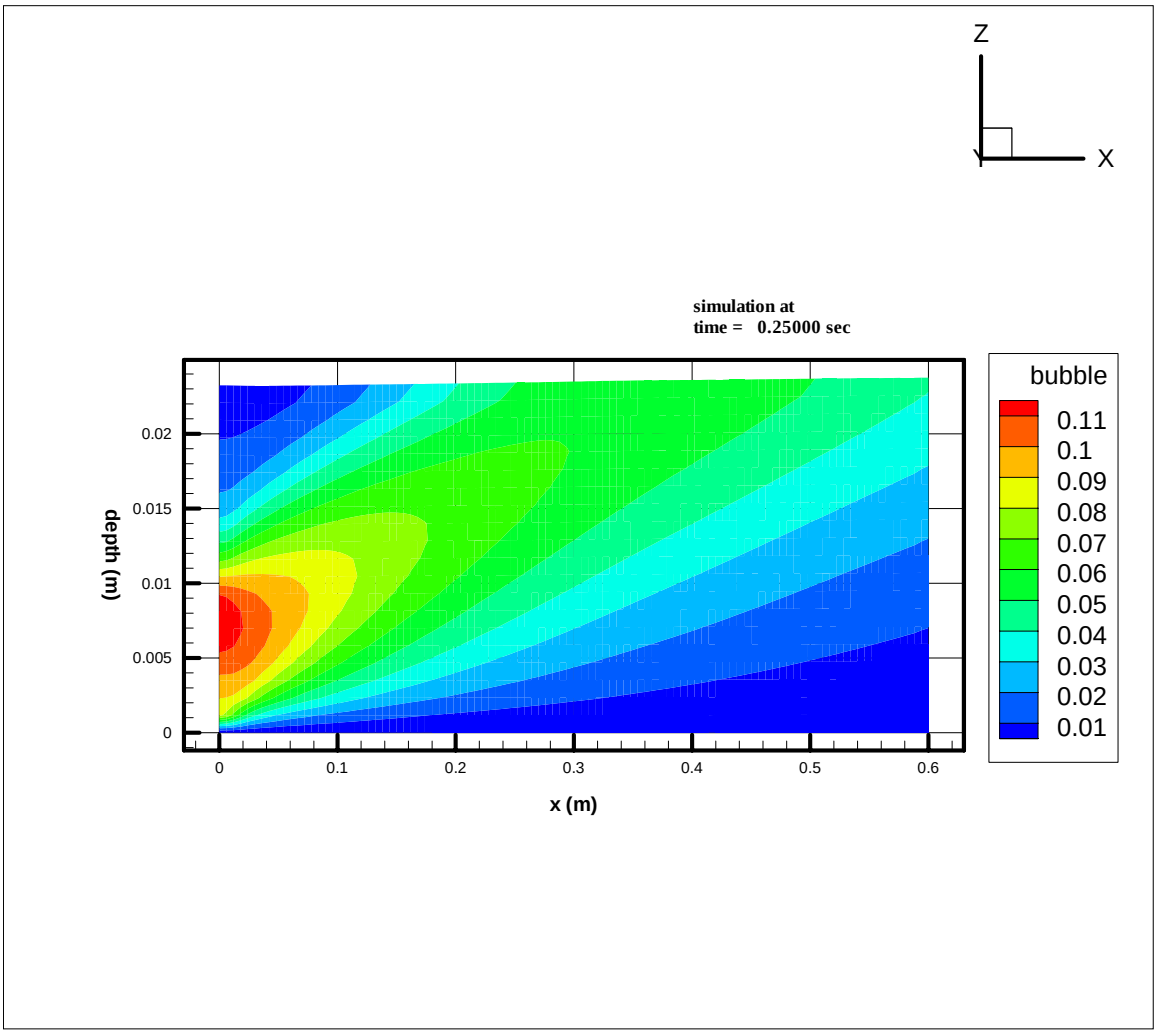

Figure 5-15: Sample numerical result of air concentration profile at steady state for bubble diameter of *0.75 mm.*

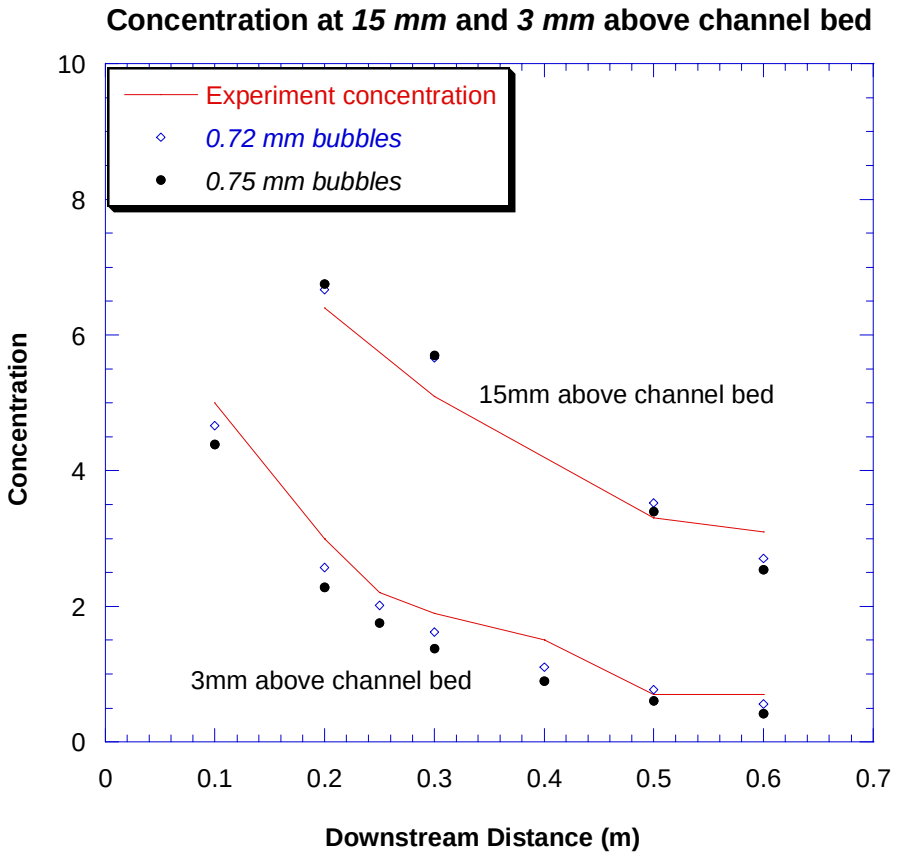

Figure 5-16: Modeled concentration of different diameter sizes and experimental data from *3 mm* and *15 mm* above channel bed.

Table 5-3: Data from Zarrati's experiment with water velocity *4.3 m/s*, medium air input, and a flow depth of *2.25 cm*.

		Experimentally observed concentration(%)						
		Distance downstream (*m*)						
elev from bed (*m*)	initial section	*0.1*	*0.2*	*0.25*	*0.3*	*0.4*	*0.5*	*0.6*
0.017			*6.9*					
0.016				*6.9*				
0.015			*6.4*		*5.1*		*3.3*	*3.1*
0.014	*2.4*	*6.9*						
0.013			*6.5*	*5.9*				
0.012					*5*	*4.3*	*3.1*	*2.2*
0.011	*6.8*	*7.9*	*6.2*					
0.01				*5*				
0.009	*11.5*	*7.8*	*5.4*		*4.3*	*3.7*	*2.4*	*1.8*
0.007	*12.4*	*7.1*						
0.006			*4.6*	*3.7*	*3.4*	*2.5*	*1.4*	
0.005	*10.8*	*6*						
0.003	*9.5*	*5*	*3*	*2.2*	*1.9*	*1.5*	*0.7*	*0.7*

Table 5-4: Air bubble concentration from numerical model for *0.75 mm* bubbles.

		Modeled concentration(%)						
		Distance downstream (*m*)						
elev from bed (*m*)	initial section	*0.1*	*0.2*	*0.25*	*0.3*	*0.4*	*0.5*	*0.6*
0.017			*6.35*					
0.016				*6.34*				
0.015			*6.75*		*5.7*		*3.4*	*2.54*
0.014	*2.4*	*7.05*						
0.013			*6.76*	*5.99*				
0.012					*4.92*	*3.6*	*2.62*	*1.91*
0.011	*6.8*	*8.3*	*6.39*					
0.01				*5.09*				
0.009	*11.5*	*8.4*	*5.73*		*3.88*	*2.69*	*1.9*	*1.35*
0.007	*12.4*	*7.8*						
0.006			*4.26*	*3.36*	*2.7*	*1.8*	*1.24*	
0.005	*10.8*	*6.53*						
0.003	*9.5*	*4.39*	*2.28*	*1.75*	*1.38*	*0.9*	*0.61*	*0.424*

Table 5-5: Percentage difference between numerical results and experiment data for air bubble concentration of *0.75 mm* diameter.

		Percentage difference (%) from experimental values						
		Distance downstream (m)						
elev from bed (*m*)	initial section	*0.1*	*0.2*	*0.25*	*0.3*	*0.4*	*0.5*	*0.6*
0.017			*-7.97%*					
0.016				*-8.12%*				
0.015			*5.47%*		*11.76%*		*3.03%*	*-18.06%*
0.014	*2.4*	*2.17%*						
0.013			*4.00%*	*1.53%*				
0.012					*-1.60%*	*-16.28%*	*-15.48%*	*-13.18%*
0.011	*6.8*	*5.06%*	*3.06%*					
0.01				*1.80%*				
0.009	*11.5*	*7.69%*	*6.11%*		*-9.77%*	*-27.30%*	*-20.83%*	*-25.00%*
0.007	*12.4*	*9.86%*						
0.006			*-7.39%*	*-9.19%*	*-20.59%*	*-28.00%*	*-11.43%*	
0.005	*10.8*	*8.83%*						
0.003	*9.5*	*-12.20%*	*-24.00%*	*-20.45%*	*-27.37%*	*-40.00%*	*-12.86%*	*-39.43%*

The maximum percentage difference from Table 5-5 was approximately *40%* and was at *3 mm* above the bed. Further away from the bottom boundary, less percentage difference was observed. Referring back to Figure 5-15, the centre of mass of the air bubbles follows a pattern that spans to about *35 cm* downstream, having an upward curving slope. Disregarding differences near the boundary, the agreement between the values of the modeled air concentration and experimental observation was reasonable within the first *30 cm* from the initial section. Within the *30 cm* span, the largest percentage difference was about *20%* of under prediction. Concentration after *30 cm* downstream are predominantly low concentration regions as the center of mass of the air bubbles have risen up to the surface. The model did reasonably well in regions of high concentrations. As the concentration decreased, the differences became more significant. This also means that the error gets more and more pronounced away from the centre of mass of the air bubble concentration.

In this simulation, only bubbles with size of *0.75 mm* were modeled. The terminal velocity of the air bubble is determined to be approximately *13.17 cm/s*. In Zarrati's experimental data, the distributions of air bubble sizes were not reported. The only mentioned sizes were that used in his numerical simulation (larger than *2 mm*, having a terminal velocity of *19 cm/s* and smaller than *2 mm,* having a terminal velocity of *12 cm/s*). In this respect, air bubbles of larger diameter will, in general have greater terminal velocity while smaller air bubbles will have smaller terminal velocities (Figure 3-2 of the terminal velocity plot). Air bubbles in the experiment were generated via a diffuser and this might have created air bubbles of various sizes. Therefore, the smaller air bubbles will have a greater residence time and hence, could have contributed to the higher concentration near the bed in the experimental data. In addition the combination of the turbulence model and under resolved grids near the wall could be the other sources of error. Near to the wall, air bubble concentration is directly influenced by the turbulent eddy viscosities and the velocities vectors. Using a low grid resolution near the wall boundary will give poor prediction of the velocity at the first cell because the use of the wall function assumes linear velocity gradient from the wall to the first cell. Consequently the modeling of the eddy viscosities which also requires the use of velocity gradient near the wall will be impacted, hence introducing mixing errors.

In view of the fact that this simulation was done with the consideration of only one air bubble diameter, some deficiency is expected when compared to experiment values. In this case, the extreme differences occurred in places with small amounts of concentration, *i.e*. near the bottom boundary and away from the center of mass. As mentioned earlier, Zarrati had hinted in his paper that his experimental observations showed the same trend; hence the *12 cm/s* terminal velocity was used for his simulation.

To further highlight the point that different size air bubbles contributed to the large percentage differences, the results from the simulation of *0.72 mm* diameter air bubble is tabulated. This simulation had the same configuration, but with a different size in the air bubble, giving a terminal velocity of approximately *12.12 cm/s*. The focus of this simulation was merely to demonstrate that the small air bubbles reside near the bottom boundary for a longer time, which gives higher concentration further away. Reducing the size of air bubble had little impact on the hydrodynamics. However, it improved the prediction of the modeled concentrations at the bottom boundary, by almost *50%* in some cases. Table 5-6 and Table 5-7 summarize the results. A significant improvement can be seen especially at a distance of *3 mm* above the bottom boundary.

Table 5-6: Air bubble concentration from numerical model for *0.72 mm* bubbles.

		Modeled concentration (%)						
		Distance downstream (*m*)						
elev from bed (*m*)	initial section	*0.1*	*0.2*	*0.25*	*0.3*	*0.4*	*0.5*	*0.6*
0.017			*6.27*					
0.016				*6.26*				
0.015			*6.67*		*5.66*		*3.52*	*2.71*
0.014	*2.4*	*7.01*						
0.013			*6.7*	*5.96*				
0.012					*4.96*	*3.73*	*2.79*	*2.1*
0.011	*6.8*	*8.24*	*6.37*					
0.01				*5.14*				
0.009	*11.5*	*8.34*	*5.76*		*4.01*	*2.88*	*2.1*	*1.54*
0.007	*12.4*	*7.82*						
0.006			*4.43*	*3.56*	*2.91*	*2.02*	*1.44*	
0.005	*10.8*	*6.61*						
0.003	*9.5*	*4.66*	*2.57*	*2.01*	*1.62*	*1.1*	*0.77*	*0.56*

Table 5-7: Percentage difference between numerical results and experiment data for air bubble concentration of *0.72 mm* diameter.

		Percentage difference (%) from experimental values						
		Distance downstream (*m*)						
elev from bed (*m*)	initial section	*0.1*	*0.2*	*0.25*	*0.3*	*0.4*	*0.5*	*0.6*
0.017			*-9.13%*					
0.016				*-9.28%*				
0.015			*4.22%*		*10.98%*		*6.67%*	*-12.58%*
0.014	*2.4*	*1.59%*						
0.013			*3.08%*	*1.02%*				
0.012					*-0.80%*	*-13.26%*	*-10.00%*	*-4.55%*
0.011	*6.8*	*4.30%*	*2.74%*					
0.01				*2.80%*				
0.009	*11.5*	*6.92%*	*6.67%*		*-6.74%*	*-22.16%*	*-12.50%*	*-14.44%*
0.007	*12.4*	*10.14%*						
0.006			*-3.70%*	*-3.78%*	*-14.41%*	*-19.20%*	*2.86%*	
0.005	*10.8*	*10.17%*						
0.003	*9.5*	*-6.80%*	*-14.33%*	*-8.64%*	*-14.74%*	*-26.67%*	*10.00%*	*-20.00%*

Figure 5-17 and Figure 5-18 show the propagation of the air bubbles through the domain. As earlier mentioned, the small wave can be seen moving at the surface near the leading edge of the bubble concentration. Figure 5-17 and Figure 5-18 shows the numerical results for air bubble concentration at *t=0.06 sec* and at *t=0.3 sec*. Figure 5-18 can be used to compare with Figure 5-15 to visually see the difference in the concentration contours of the two different sizes of air bubbles.

The smaller diameter air bubbles resulted in higher concentration at *3 mm* above the channel bed throughout the length of the channel compared to that of the larger air bubbles. The overall percentage difference is also improved. Zarrati did mentioned in his paper that the smaller size air bubbles tend to occupy the region where concentration is lower than *5%*, and these are precisely the regions where the percentage differences were large in the initial simulation with *0.75 mm* diameter air bubbles. Without accurate size distributions of the air bubbles, the model prediction can only be an estimate.

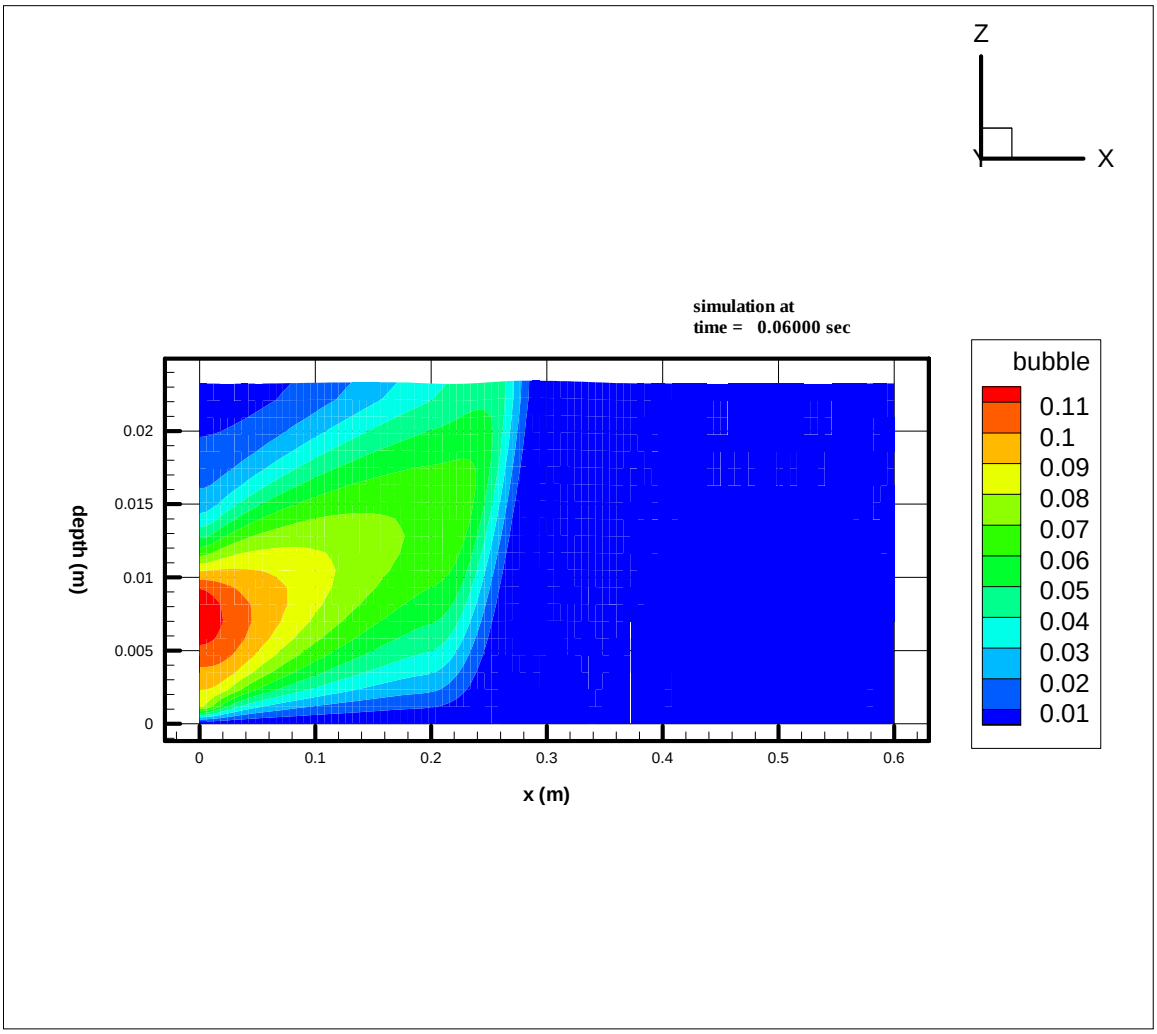

Figure 5-17: Sample numerical result of air concentration profile at *t=0.06 sec* for bubble diameter of *0.72 mm*.

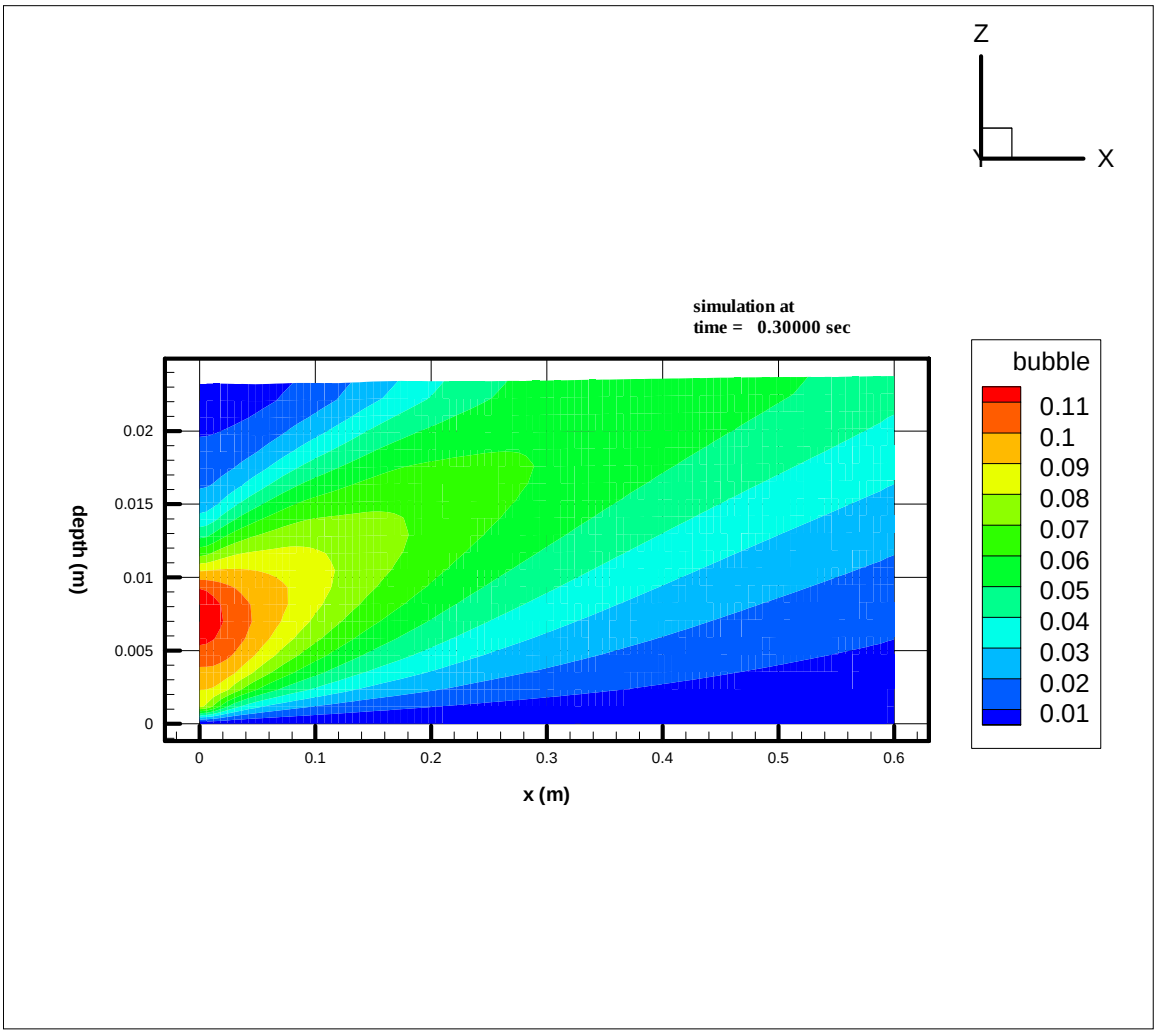

Figure 5-18: Sample numerical result of air concentration profile at steady state for bubble diameter of *0.72 mm.*

The results obtained from the *0.75 mm* and *0.72 mm* diameter air bubbles simulation suggested that the model was, to an extent, sensitive to the terminal rise velocity or air bubble diameter. A simple sensitivity analysis was carried out. The sensitivity analysis was based on Zarrati's reported terminal velocities in his numerical model. In Zarrati's numerical model a terminal velocity of *12 cm/s* was used for the smaller air bubbles, while *19 cm/s* for the larger size air bubbles. The same reference velocities are used for relative comparison. The reference velocities were varied by *±10* percent ($V_\infty \pm 0.1V_\infty$) and plotted on the same graph for comparison. The details were summarized in Table 5-8. The corresponding differences between the bubble diameter sizes are approximately *5.1%* (for $V_\infty + 0.1V_\infty$) and *-5.6%* (for $V_\infty - 0.1V_\infty$) from that of the reference velocity of *12 cm/s*. For the reference velocity of *19 cm/s*, the percentage differences of the bubble diameter sizes are approximately *6%* (for $V_\infty + 0.1V_\infty$) and *-6%* (for $V_\infty - 0.1V_\infty$). Plots of concentration contours were shown in Figure 5-19 and Figure 5-20. A common trend was observed in these figures. At the inflow boundary, there was no different in air bubbles concentration profile, as the boundary information was specified. The air bubble concentrations became increasingly sensitive to the terminal velocities as the distance downstream from the inflow boundary increased. Air bubbles generally have the natural tendency to rise while being transported downstream. Therefore, the terminal velocity becomes a vital part of the air bubble transport model. A greater terminal velocity gives faster rise, while smaller terminal velocity is associated with slower rise for the air bubble concentration and this relationship holds true all the way through the length of the channel in the simulations. Strictly speaking, the differences in concentration of the reference velocity, V_∞ and that of the $V_\infty \pm 0.1V_\infty$ (in Figure 5-19 and Figure 5-20) gradually becomes greater downstream because the difference of concentration downstream is a function of cumulative differences from upstream of the channel, due to the different terminal rise velocity.

Table 5-8: Details of simulations for sensitivity analysis

	12 cm/s reference velocity		*19 cm/s* reference velocity	
	terminal velocity	bubble diameter	terminal velocity	bubble diameter
$(V_\infty + 0.1V_\infty)$	*13.2 cm/s*	*0.7515 mm*	*20.9 cm/s*	*0.975 mm*
V_∞	*12 cm/s*	*0.715 mm*	*19 cm/s*	*0.92 mm*
$(V_\infty - 0.1V_\infty)$	*10.8 cm/s*	*0.675 mm*	*17.1 cm/s*	*0.865 mm*

Sensitivity of the concentration profiles attributable to the different terminal velocities will therefore be more visible further away from the inflow boundary. Although not obvious earlier in the discussion, the percentage differences in concentration shown in Table 5-5 and Table 5-7 exemplify this effect, in addition to other causes mentioned. This further reinforces the fact that the numerical model is sensitive to the terminal velocity and requires reasonable data regarding bubble diameter (which corresponds to the correct terminal velocity) for accurate modeling of the air bubbles migration in the domain.

Reasonable agreement of air bubble concentration was obtained near the center of mass of the air bubble path. However, away from the center of mass, there were justifiable differences in the modeled concentration and the experimental data. A simple test of varying the bubble diameter resulted in improved prediction of air bubble concentration near the bed and elsewhere, especially in regions of lower concentrations. This can be explained by the lower terminal velocities of the bubbles which make it possible for greater amount of air bubble to be transported downstream.

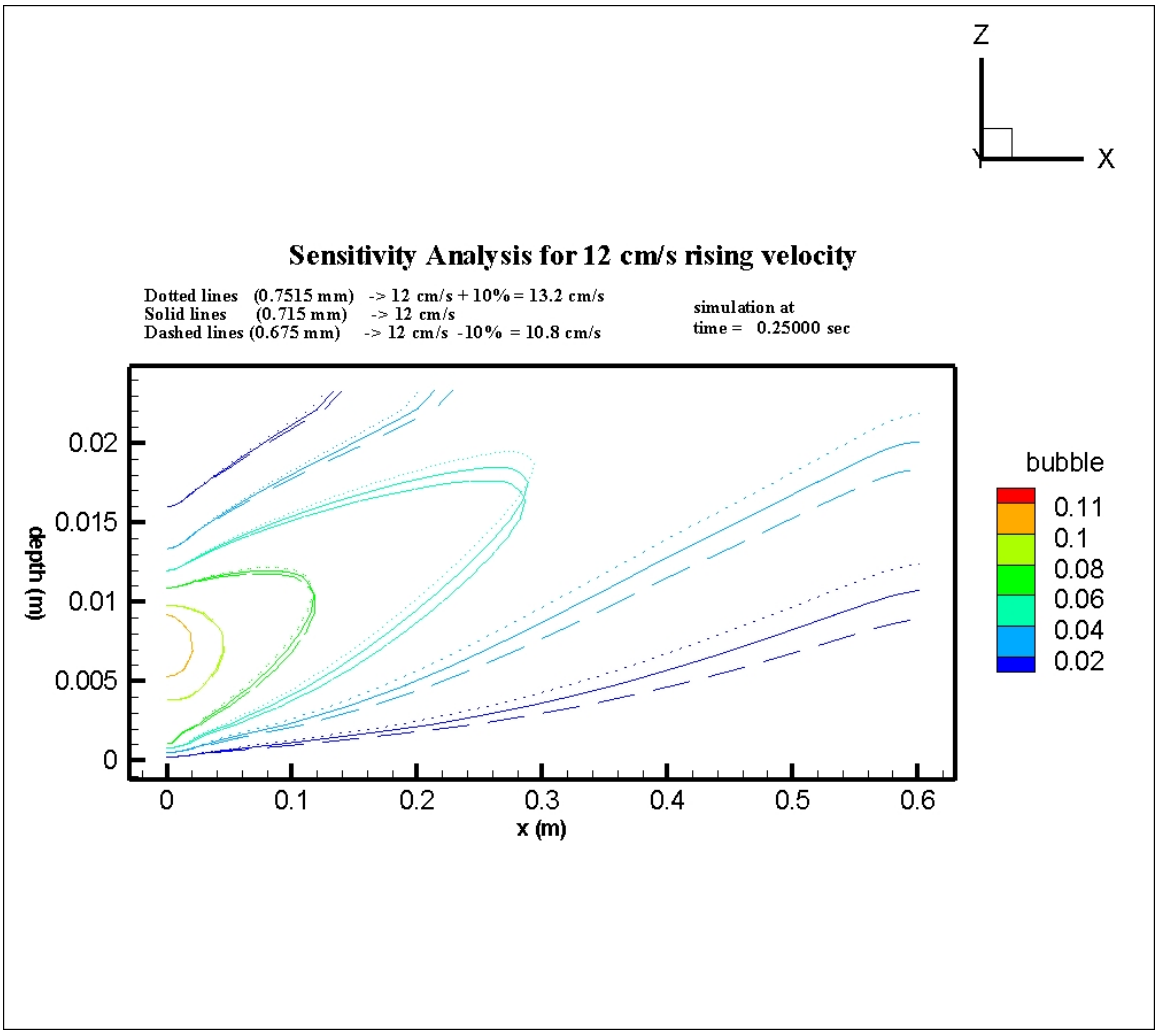

Figure 5-19: Sensitivity analysis for terminal velocity *12 cm/s*.

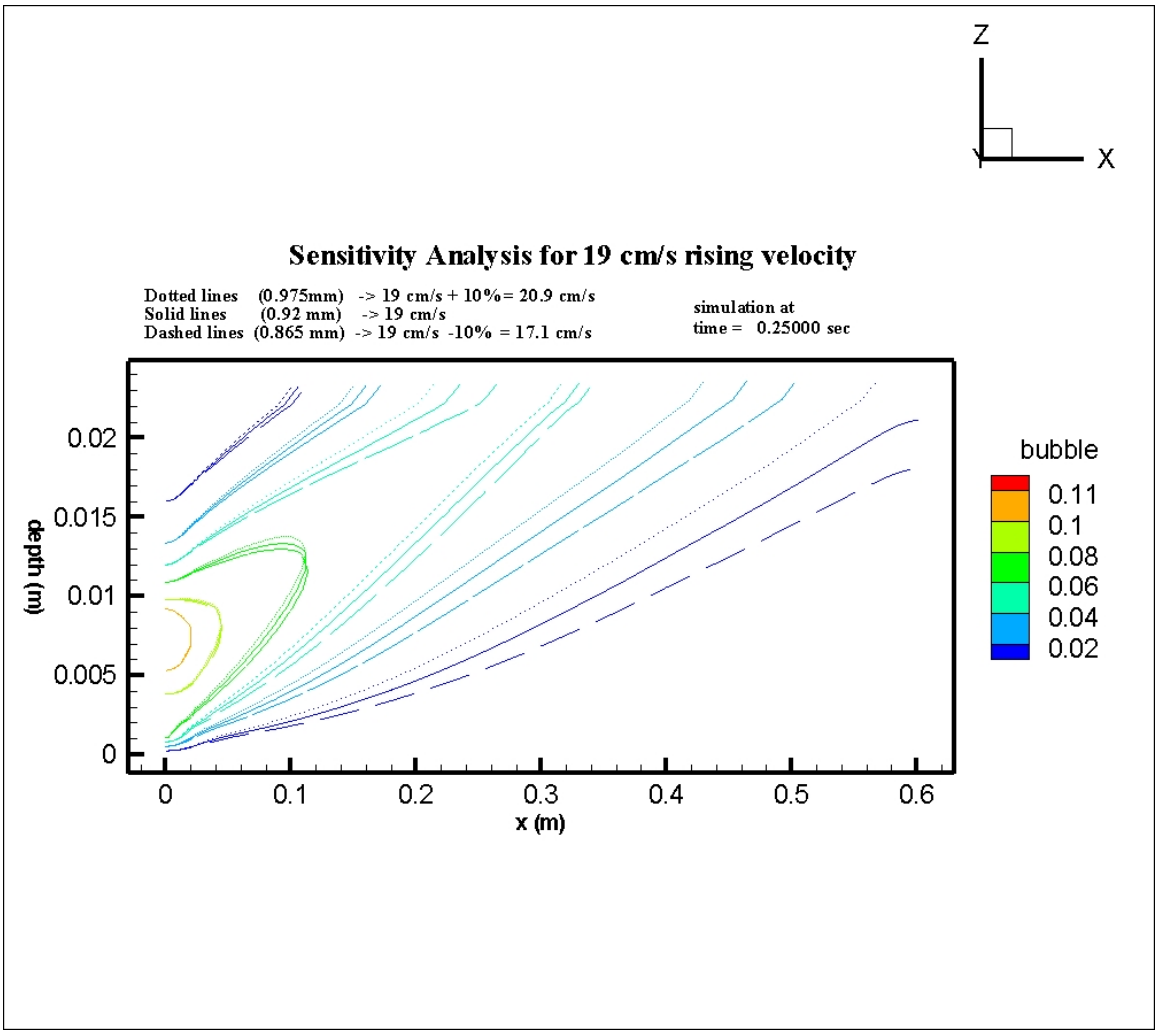

Figure 5-20: Sensitivity analysis for terminal velocity *19 cm/s*.

5.8 - *Plunging Jet Simulation with and without Air Bubbles*

The hydrodynamics and bubble transport model have been verified and showed reasonable agreement to analytical and experimental results. The next step is to simulate a hypothetical plunging jet scenario.

The simplest case to setup was to assume that the tank is a sloped channel with walls on the sides. The initial still water depth should be relatively deep compared to the potentially generated waves. Literature regarding solitary wave ([14], [20], [70]) suggested that the ratio of amplitude of a wave to the still water depth, A_{wave}/D_0, should be less than 0.4 to avoid erroneous wave propagation. Therefore, care was taken to ensure that the waves generated near the plunging section would not be a major source of error, which could lead to unphysical results or instability. Grid sizes were chosen having industry application in mind. Fine grids could be selected for academic purposes due to the availability of super computers, but for industry applications this may not be so. Thus, a relatively coarse grid was utilized. The dynamic SGS model was used for this simulation.

The hypothetical domain was chosen to have a dimension of $4\ m \times 1.4\ m \times 1.6\ m$ in the x, y and z direction, respectively, which is divided into $40 \times 16 \times 25$ computational cells. A slope of $S_o=0.001$ was specified, and standard fluid properties of water used. The overfall had a width dimension of $0.7\ m$, and was situated $1\ m$ above the water surface. The flowrate of $0.21\ m^3/s$ was specified for the overfall. At the bottom of the channel, the quadratic drag law wall function (equation (2.13.7)) was used.

To check for mass conservation, a test was conducted where the channel was converted into a tank by applying the no slip boundary to all four sides. For this simulation, air bubbles were not modeled. The test showed accumulation of fluid, as expected, with the total volume after $10\ sec$ is approximately $11.07\ m^3$, while the calculated volume from the flow rate was $11.06\ m^3$. This shows that mass was conserved using the plunging jet as the inflow boundary. Results obtained from this run were shown in Figure 5-21 through Figure 5-26. The computational grid and initial jet location (high magnitude of download velocity) is shown in Figure 5-21.

The rise of the water level can be seen alongside with the velocity vectors after *10 sec* in Figure 5-22 through Figure 5-26. Since the depth in the tank increased over time, the location of the jet changed accordingly. The shift in the jet location causes the stagnation point at the bottom of the tank to shift at different times of the simulation, however only results at *t=10 sec* will be shown here. Figure 5-22 and Figure 5-23 shows the vertical velocity color contours at different cross sections. The jet location can be identified by the strong downward velocities.

In Figure 5-24, the dynamic pressure was observed to have peaked at the location of the stagnation point, at the bottom of the tank. The same behavior of the dynamic pressure was also reported by Liao *et al.* [81] in his numerical simulation of a plunge pool from a high arch dam. Velocity vectors shown in Figure 5-26 illustrates the plunging jet location, where the dense and high velocity vectors are located at the core, and is used to trace the overfall jet. To assist in locating the jet, the surface plot of the strong downward velocity vectors (*-1 m/s*) was also shown in the same figure.

As a minimum, stability criteria based on the CFL condition shown in equation (2.15.1) must be satisfied. The temporal step should also be small enough such that the Von Neumann criteria, equation (2.15.2), is not violated as discussed in section 2.15. In essence, a conservative estimate for the CFL condition is to use the magnitude of the impinging velocity, $|u_{jet}|$ for all three coordinate directions and ensure that the chosen time step is appropriate. For this simulation, a time step of $\Delta t=1 \times 10^{-3}$ was chosen. Several trial runs with the chosen time step found that the stability conditions (equation (2.15.1) and (2.15.2)) were satisfied.

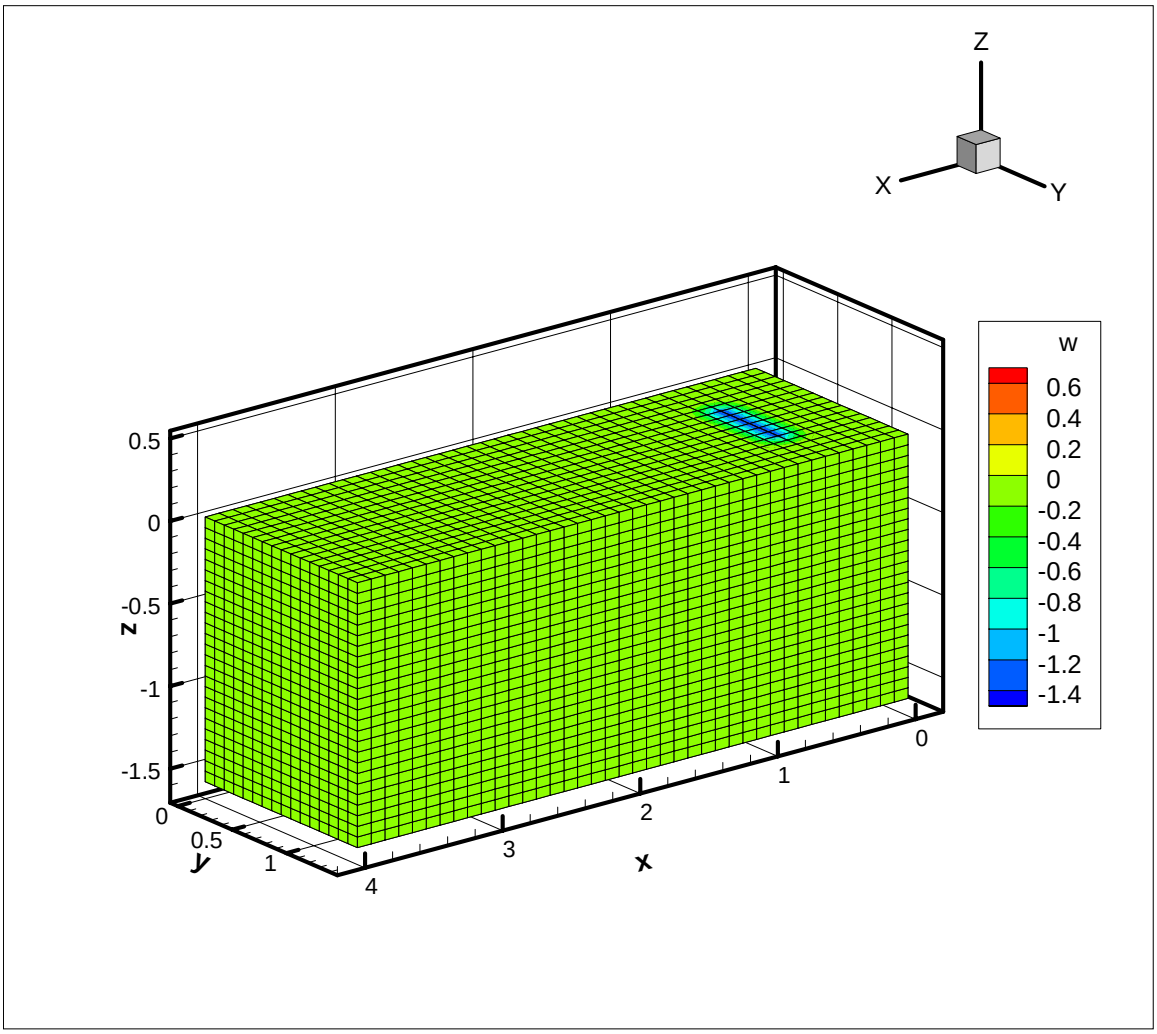

Figure 5-21: Computational grid and overfall jet location for tank simulation.

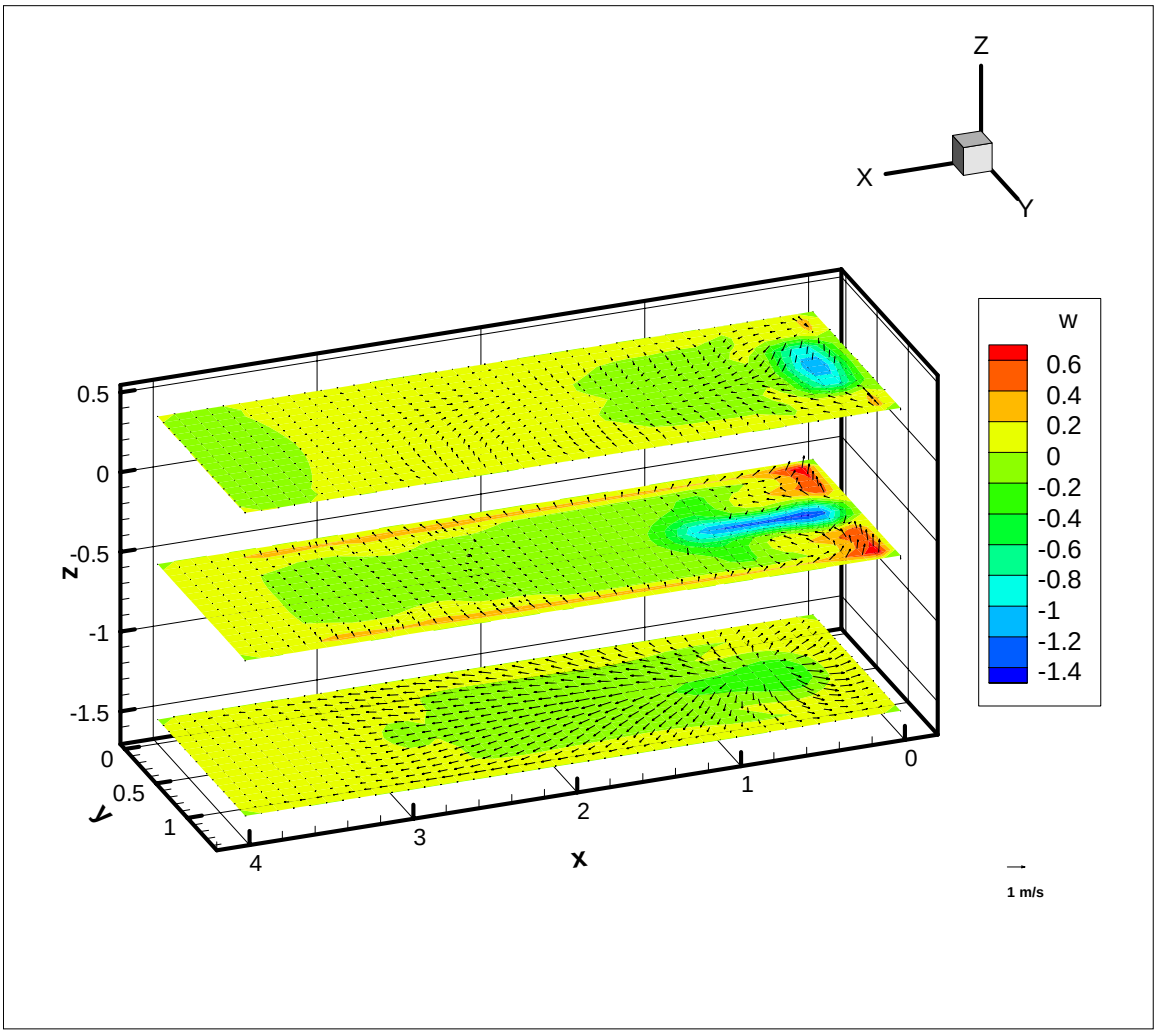

Figure 5-22: Vertical velocity color contours at bottom, mid section and near surface planes.

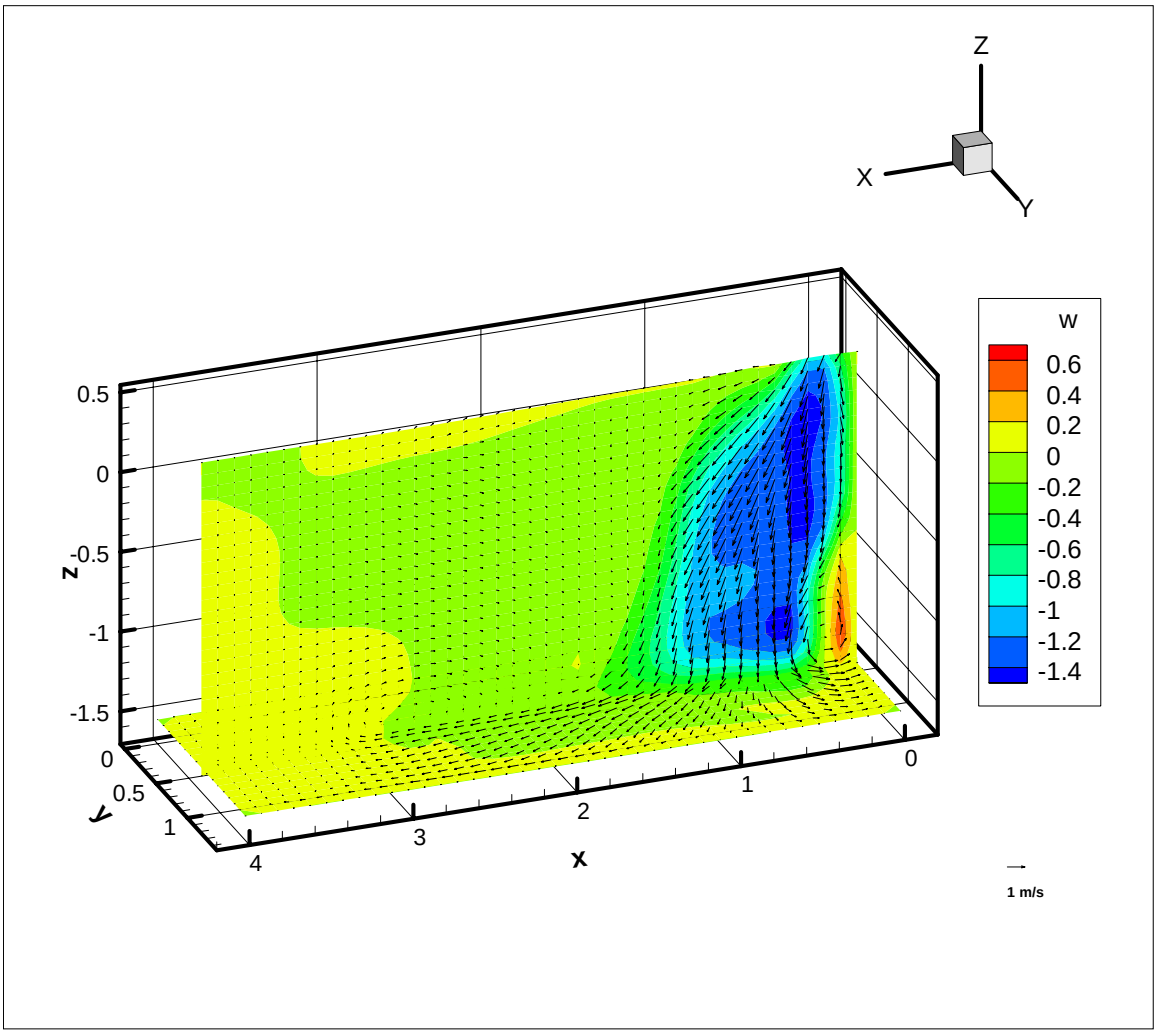

Figure 5-23: Vertical velocity color contours and velocity vectors for *X*-*Z* plane cross section at center of tank and *X*-*Y* plane cross section at bottom of tank.

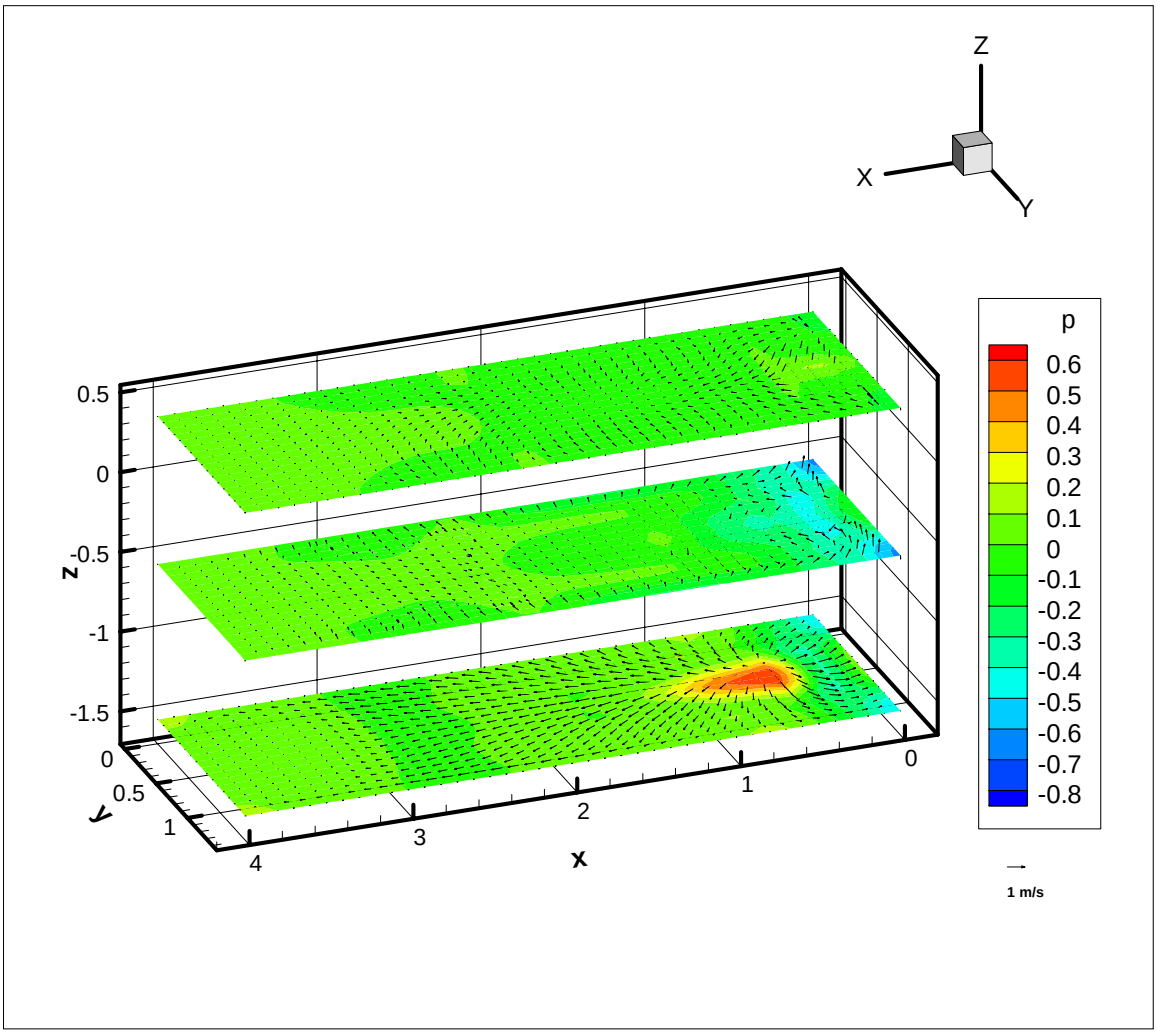

Figure 5-24: Dynamic pressure color contours and velocity vectors at bottom, mid section and near surface planes for *t=10 sec.*

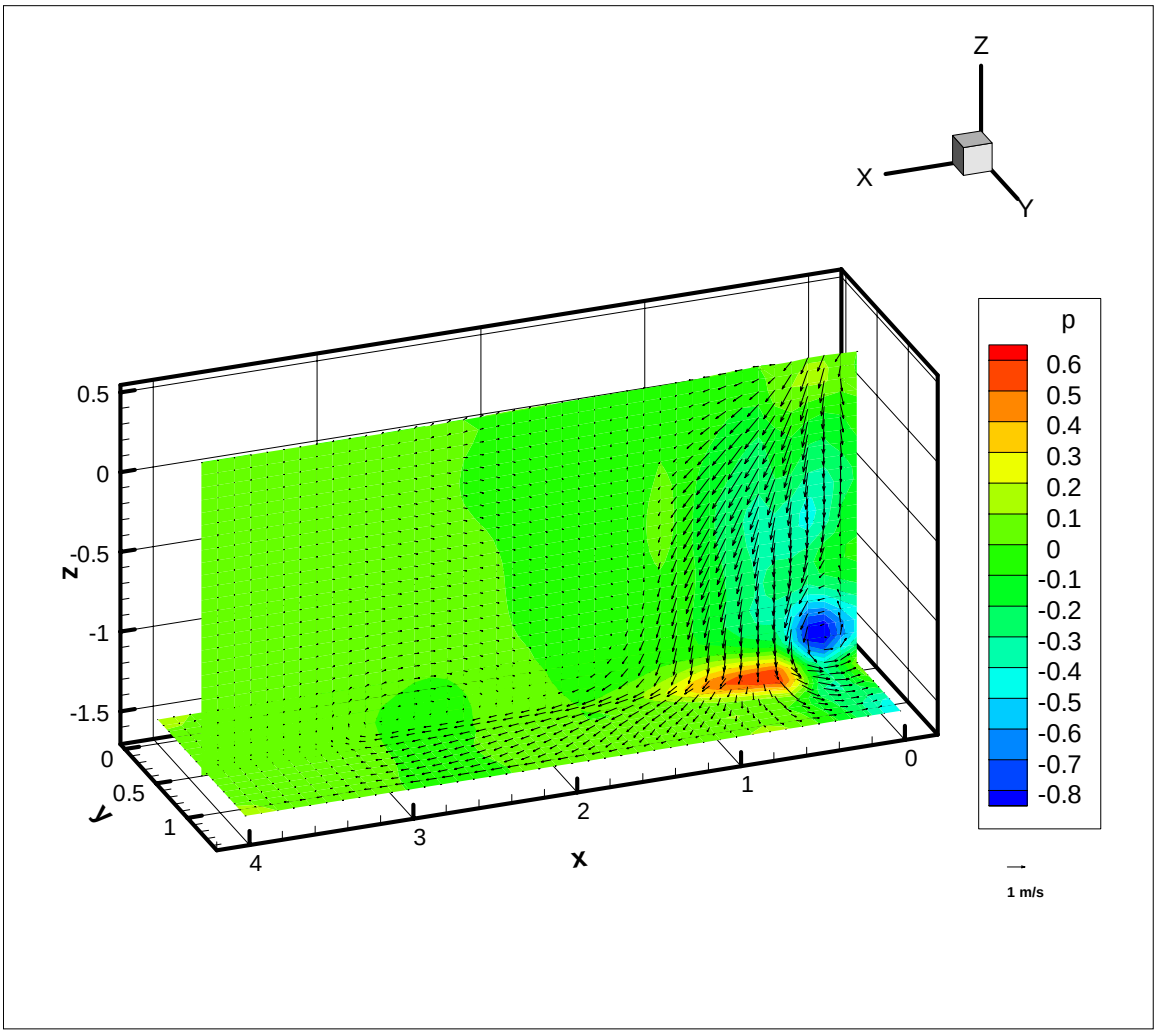

Figure 5-25: Dynamic pressure color contours and velocity vectors for *X*-*Z* plane cross section at center of tank and *X*-*Y* plane cross section at bottom of tank.

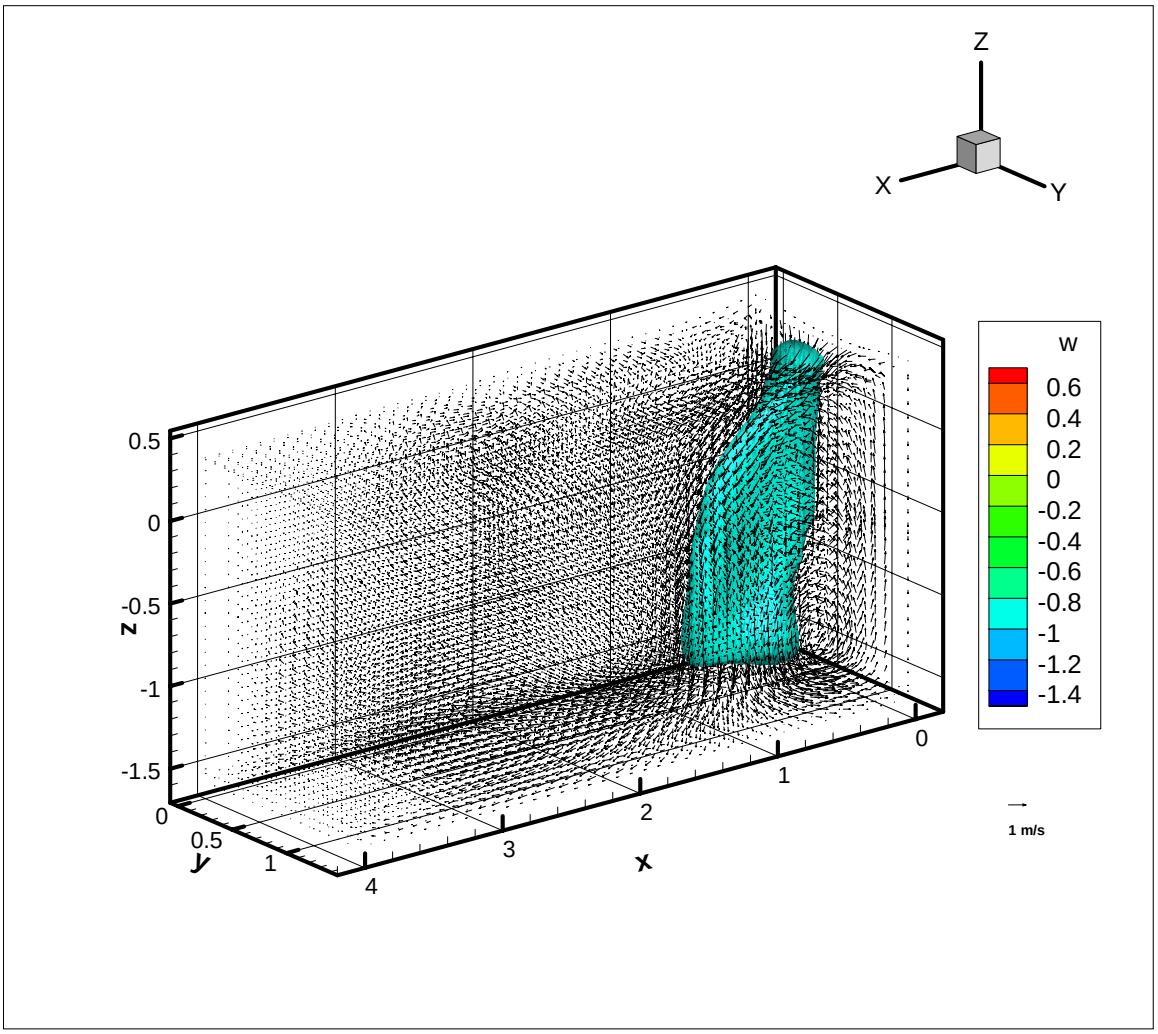

Figure 5-26: Velocity vectors and surface plot of *-1 m/s* vertical velocity in the domain of the tank at *t=10 sec.*

For cases with outflow boundary, a fixed depth downstream boundary was assumed such that the depth would be greater than the critical and normal depth and would be maintained throughout the whole simulation period. A second run was carried out for a channel with a fixed depth boundary for the outflow. This simulation was done with the intention of establishing a steady state condition where the global flux transfer is zero. In the industry, limited volume in pump stations often times requires that the global mass transfer be small if not zero, and will be controlled by pump operations. Therefore, the intention on this simulation was precisely that, but instead of pumps, the outflow boundary was set as a control. The size of air bubbles chosen for this simulation was *1 mm* in diameter. The flow, geometry, spatial and temporal time step sizes was kept the same as that in the previous simulation. At the start of the simulation, the air bubble concentration generated by the plunging jet was about *5.22%* and the jet impingement made a 77° angle with the surface of the water. The simulation was carried out for *100 sec*. Changes in depth was minute throughout the simulation.

In Figure 5-27, the velocity vectors and air bubble concentration at the center of the channel was shown. The recirculation zone described in Hadjerioua *et al.* [63] can also be identified. The recirculation zone can be thought of as the energy dissipation region where the high momentum from the jet is dissipated in the form of rotating fluid. This zone can be distinguished by the velocity vectors being re-circulated back to the direction of the plunging jet. In addition, the recirculation of the flow back to the jet also brought along some air bubbles which loops back to the near vicinity of the jet.

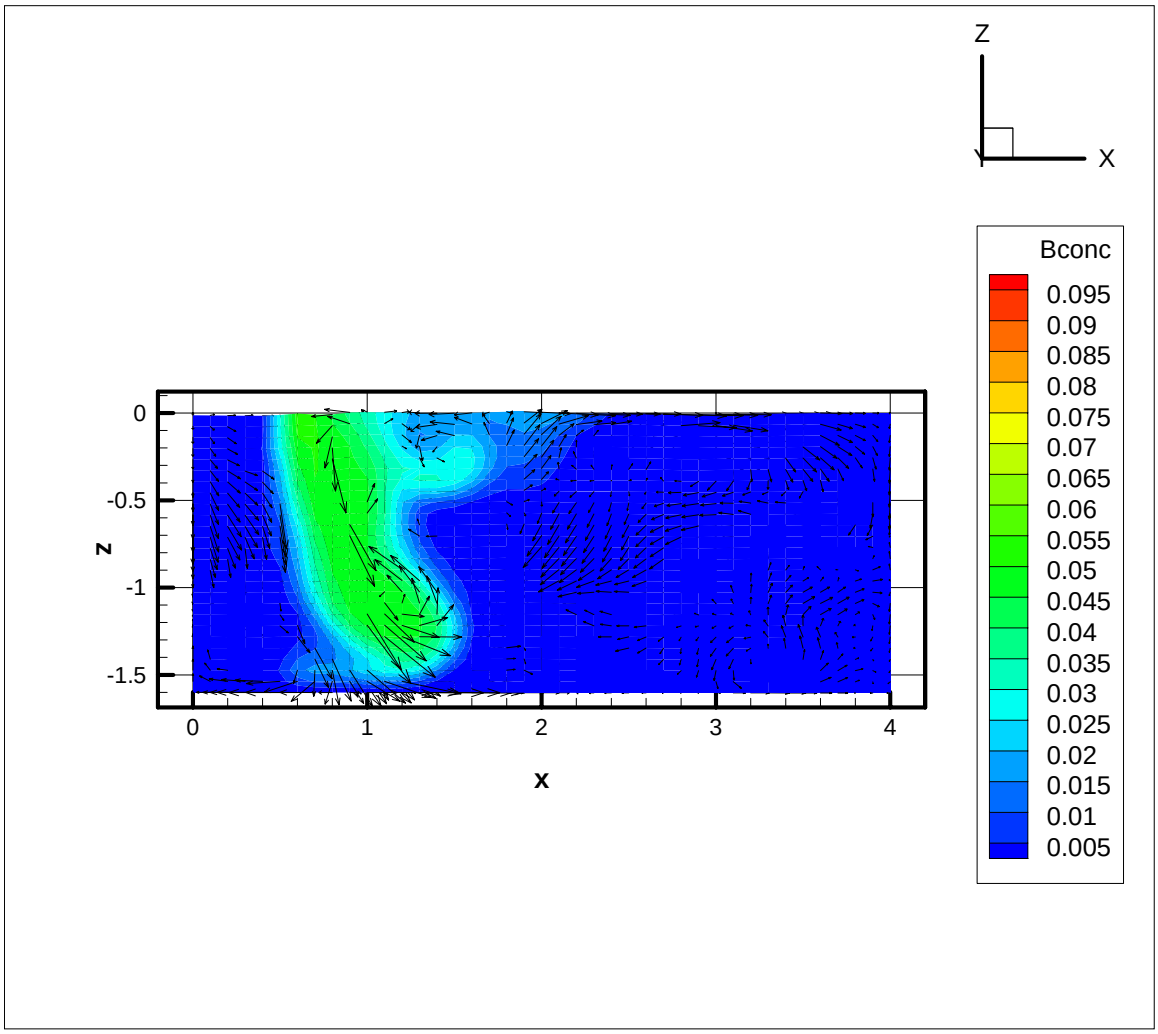

Figure 5-27: *X-Z* plane channel center cross section plot of plunging jet simulation at *t=100 sec.* Cross section taken at center of channel.

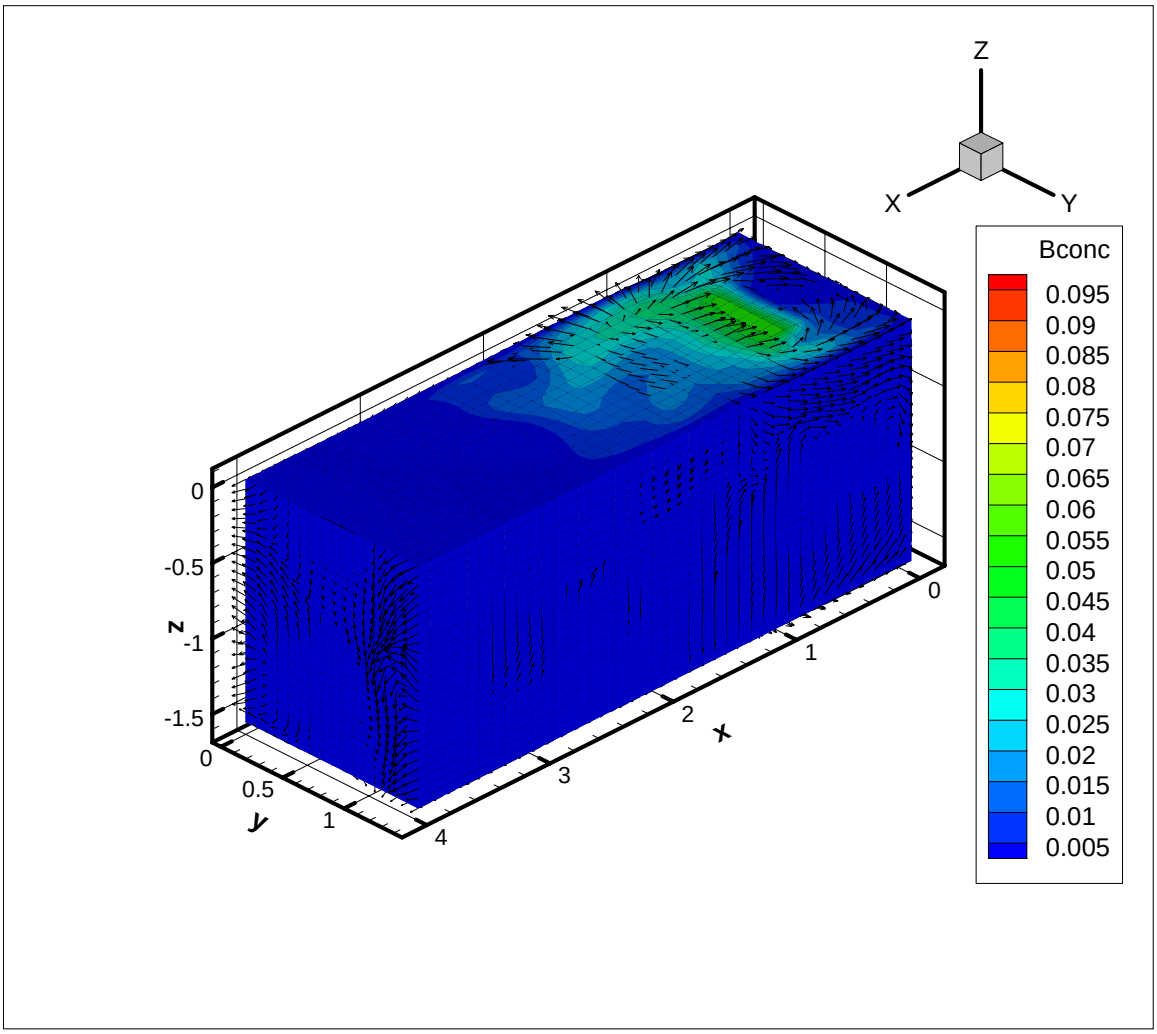

Figure 5-28: Near surface bubble concentration for plunging jet simulation at *t=100 sec.*

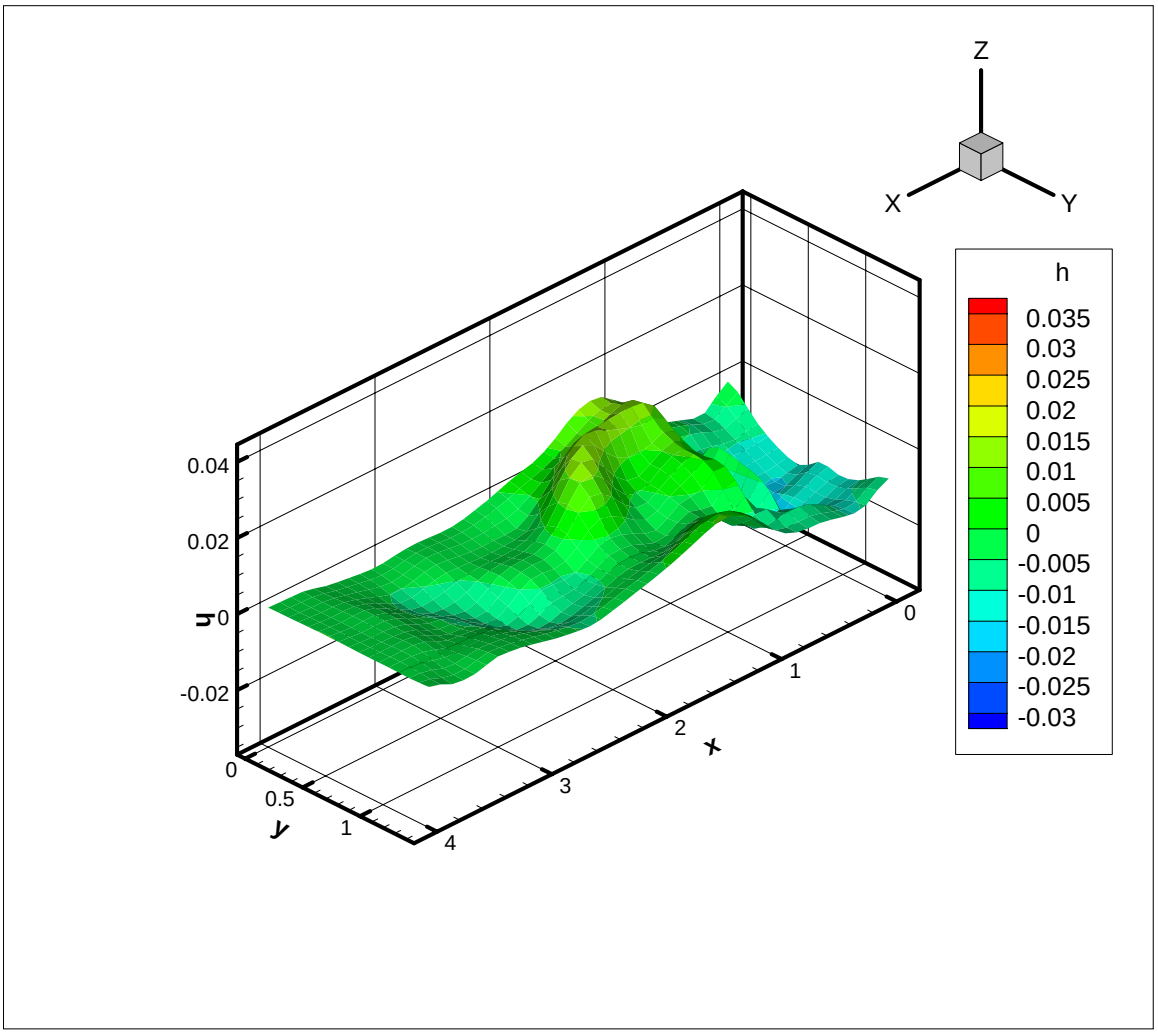

Figure 5-29: Free surface fluctuations for plunging jet simulation at *t=100 sec.*

Figure 5-28 shows the air bubble concentration near the free surface. The free surface deviation cannot be clearly seen in this figure and therefore, Figure 5-29 is a magnified view of the free surface deviation from its mean. As discussed in Chapter 3, the rising air bubble will add momentum to the surrounding fluid. In regions of higher bubble concentrations, away from the jet, the free surface increases. This suggests that the source terms due to the air bubbles are affecting the hydrodynamics of the fluid.

Maximum dynamic pressure occurs at the bottom bed around the stagnation point, similar to the tank simulation. It is interesting to note that the minimum pressure happened to be at the recirculating zone, closely situated to the stagnation point, and was more apparent near the wall boundaries. This behavior was first reported by Borghei [12] in his experiment for a circular vertical plunging jet. In experiments conducted by Borghei, the negative hydrodynamic pressure had more effect at the side walls due to the recirculation of the jet flow, and occurred near the vicinity of the jet core. The experimental observation agrees well with the result obtained from this simulation, as the flow recirculation seen in Figure 5-30 was associated with the large negative hydrodynamic pressures. The high dynamic pressure gradient between the two areas relates to a high rate of change of kinetic energy. This further implies the influence of turbulence. Figure 5-31 shows that near the recirculation zone negative eddy viscosities were predicted by the dynamic SGS model while the eddy diffusivity is greatest along the jet column.

The surface plots of the air bubble concentrations are shown in Figure 5-32 through Figure 5-36. The greater air concentration resides near the plunging jet and as the distance from the jet increases, the concentration decreased progressively. Looking at the *5%* concentration, the quantitative distribution of the bubble concentration agrees well with experiment observations of the air bubble downward diffusion cone by Chanson [24], illustrated in Figure 5-37. Instead of the two cones as illustrated in the figure, the overfall jet will form a rectangular like volume of high concentration. The magnified view of the diffusion cone is shown in Figure 5-38.

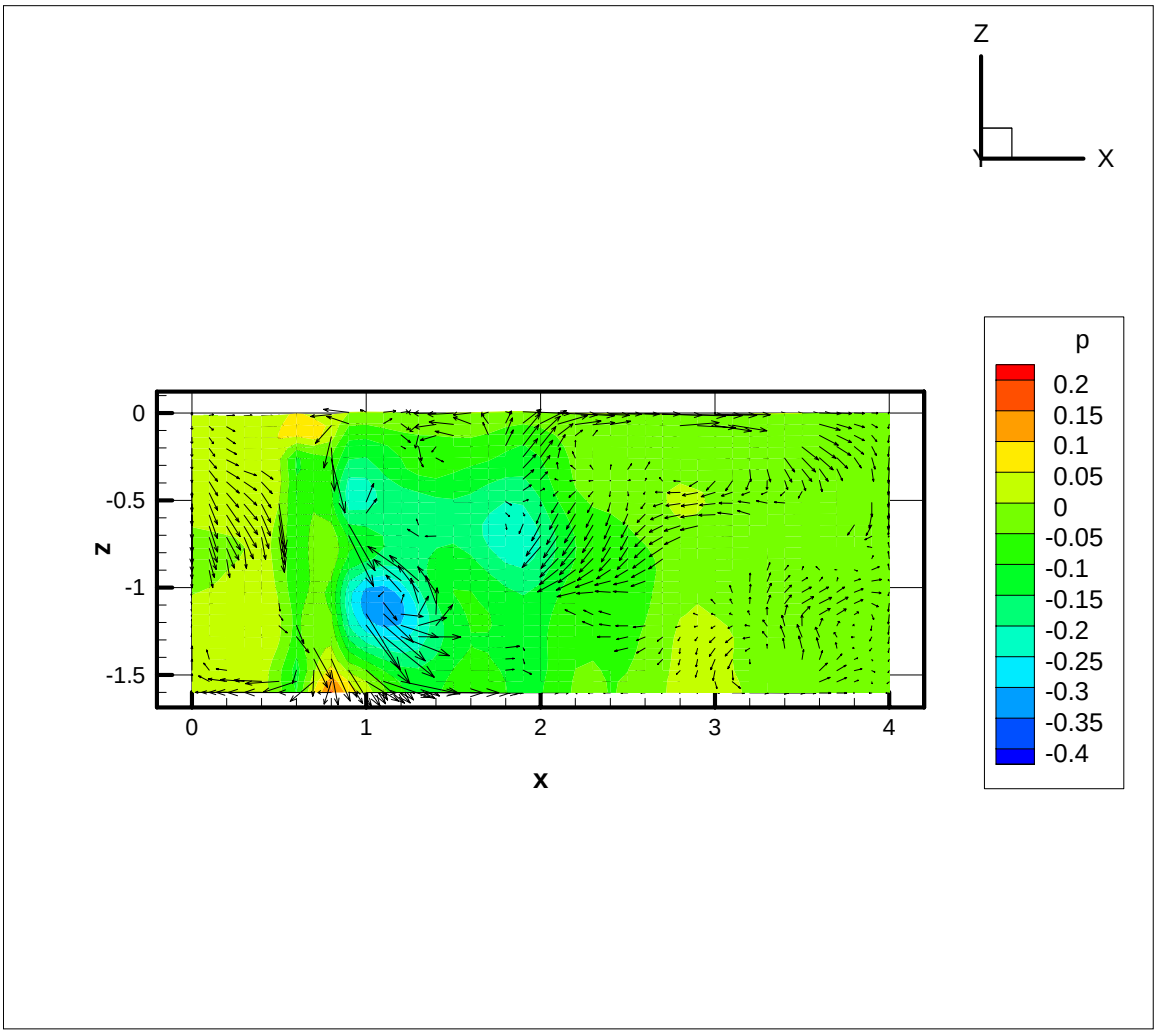

Figure 5-30: X-Z plane channel center cross section plot of pressure for plunging jet simulation at *t=100 sec*.

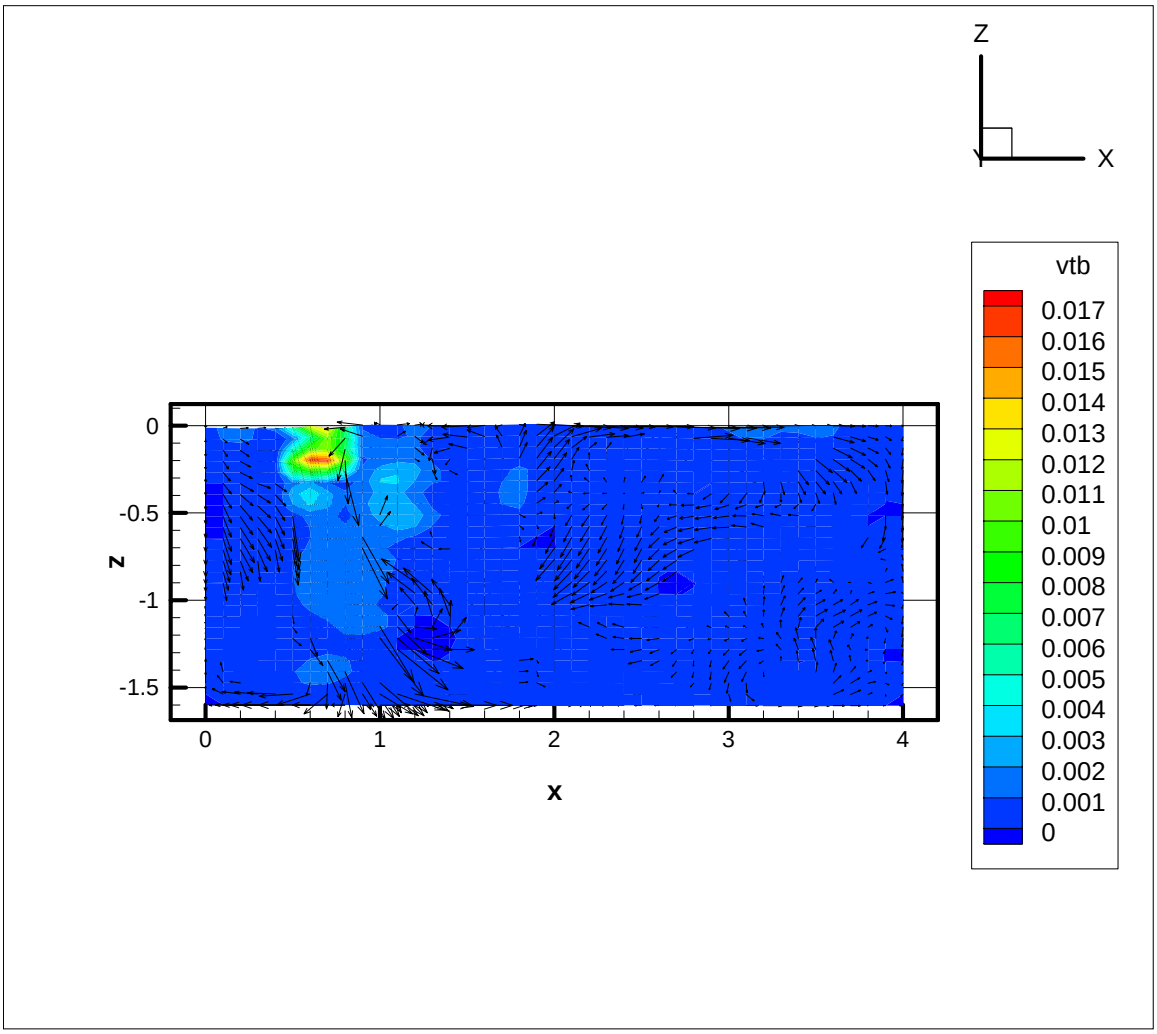

Figure 5-31: *X-Z* plane channel center cross section plot of eddy viscosity for plunging jet simulation at *t=100 sec.*

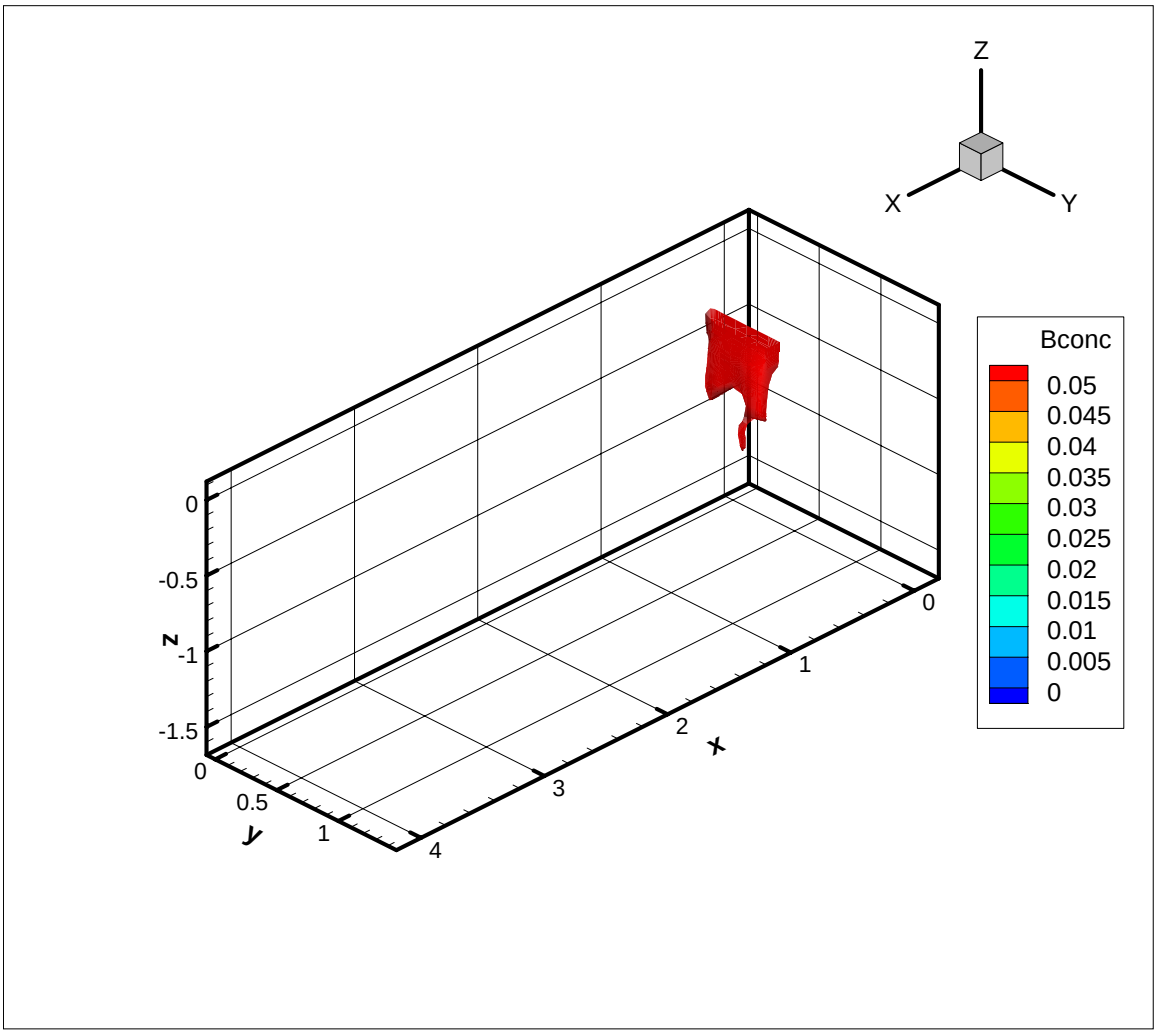

Figure 5-32: Modeled air bubble concentration of *5%.*

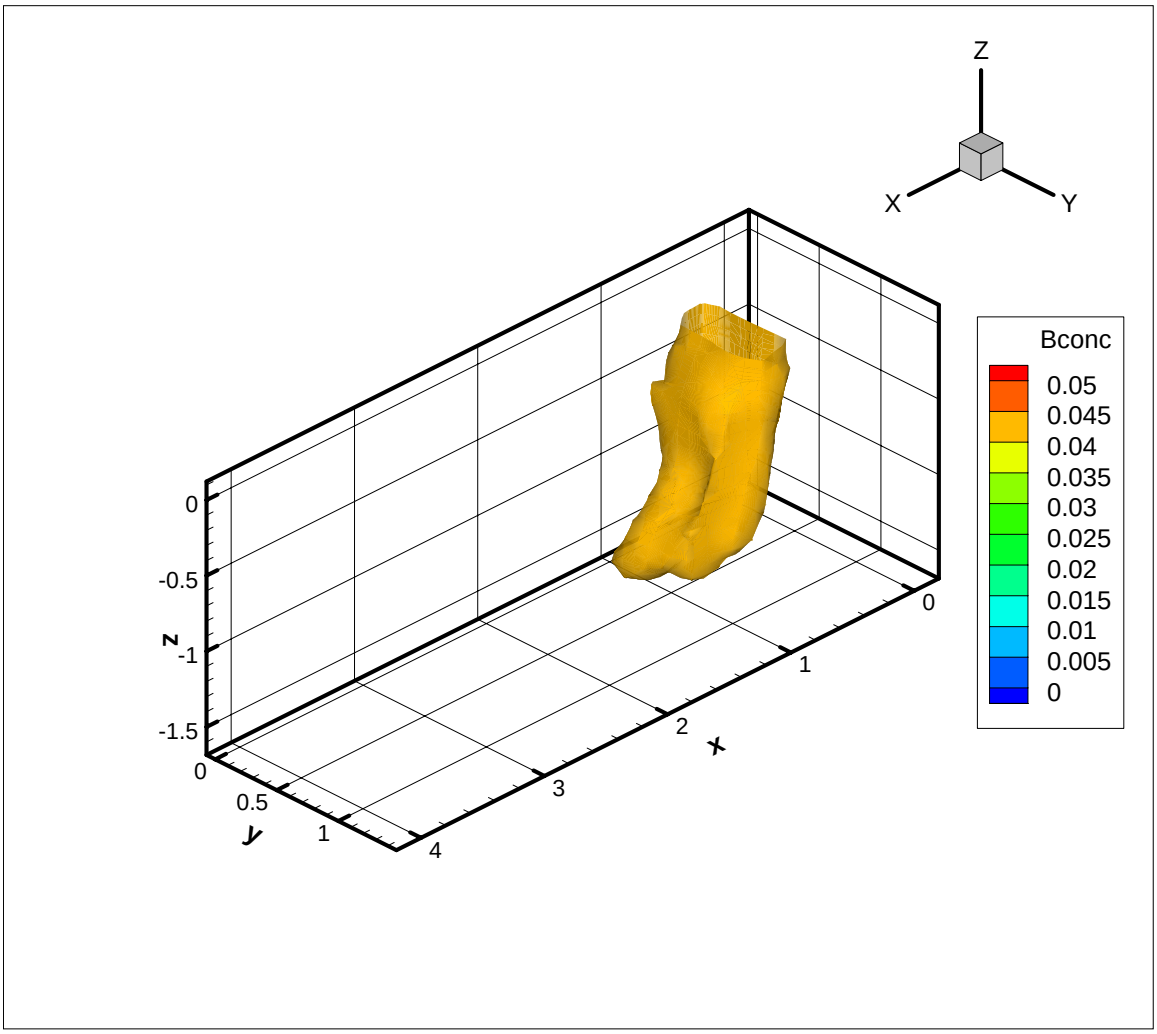

Figure 5-33: Modeled air bubble concentration of *4%.*

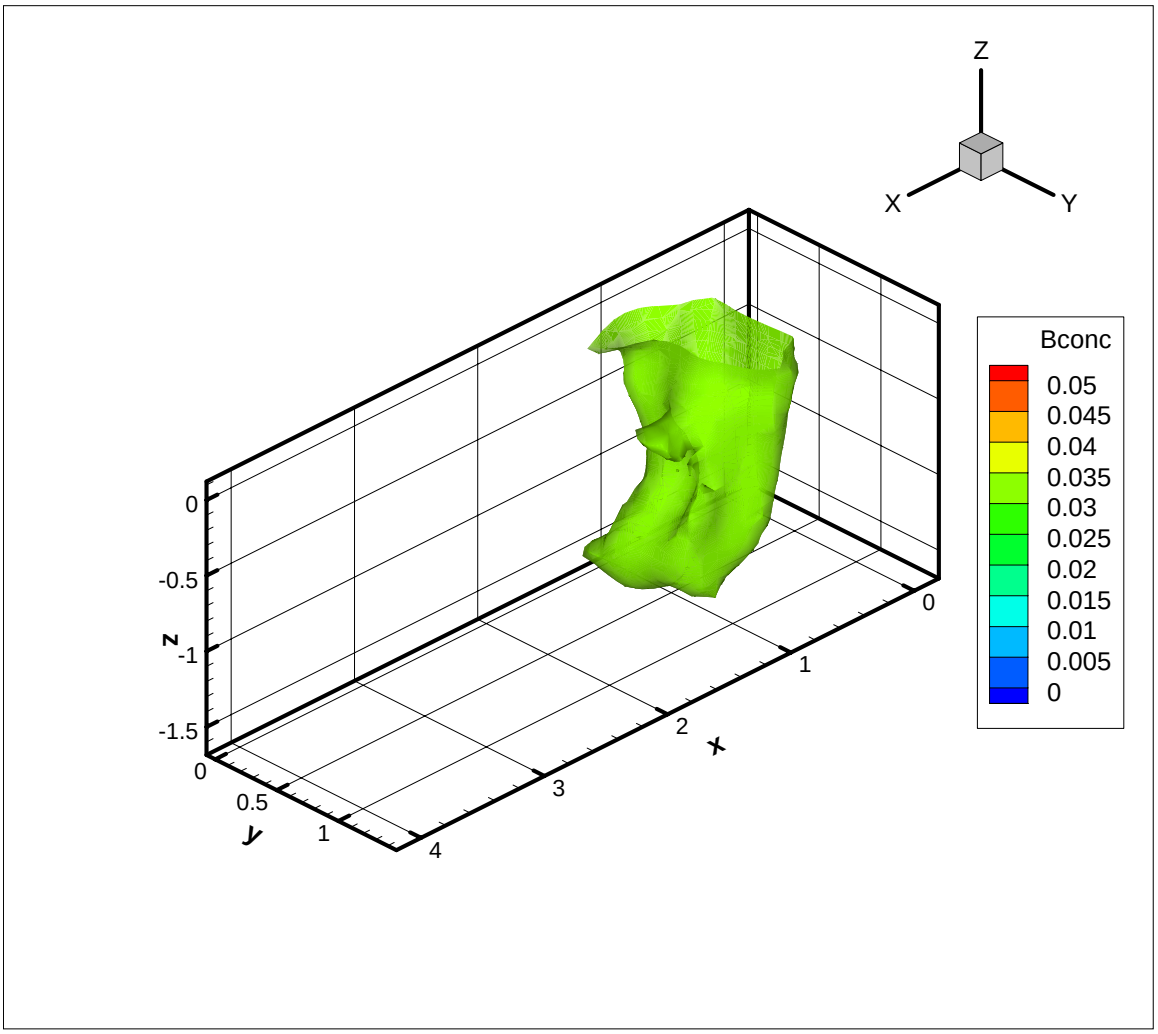

Figure 5-34: Modeled air bubble concentration of 3%.

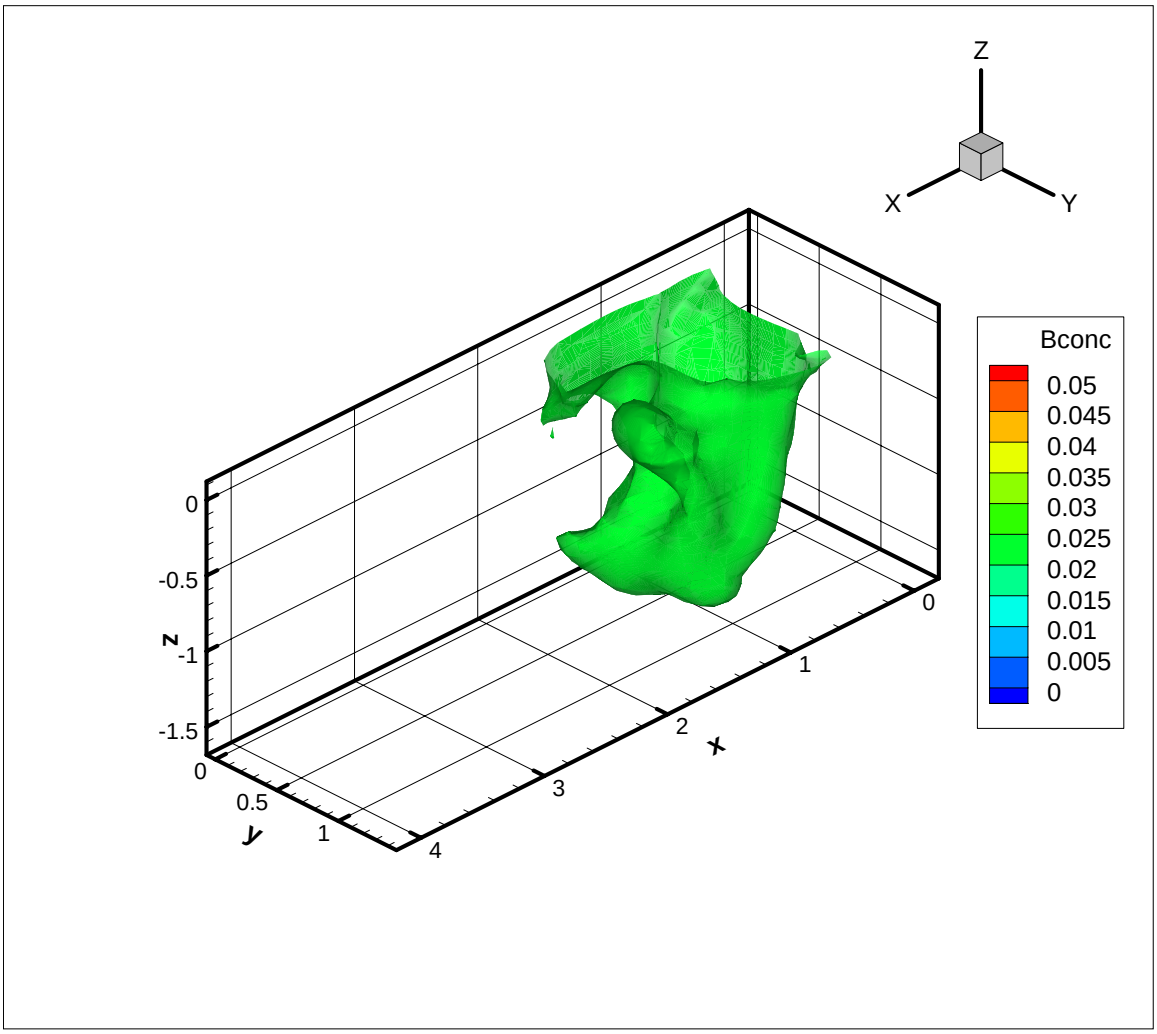

Figure 5-35: Modeled air bubble concentration of *2%.*

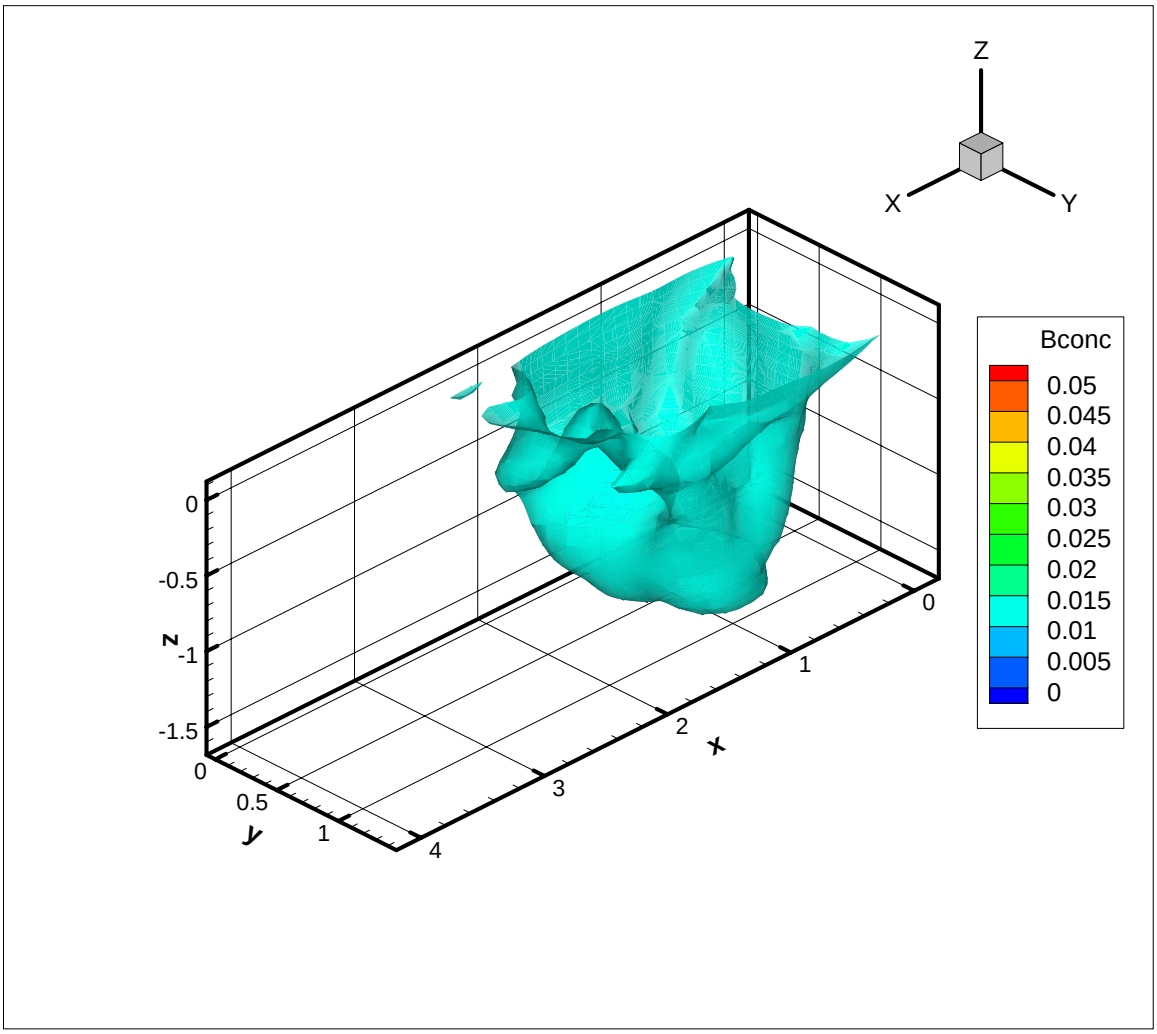

Figure 5-36: Modeled air bubble concentration of *1%.*

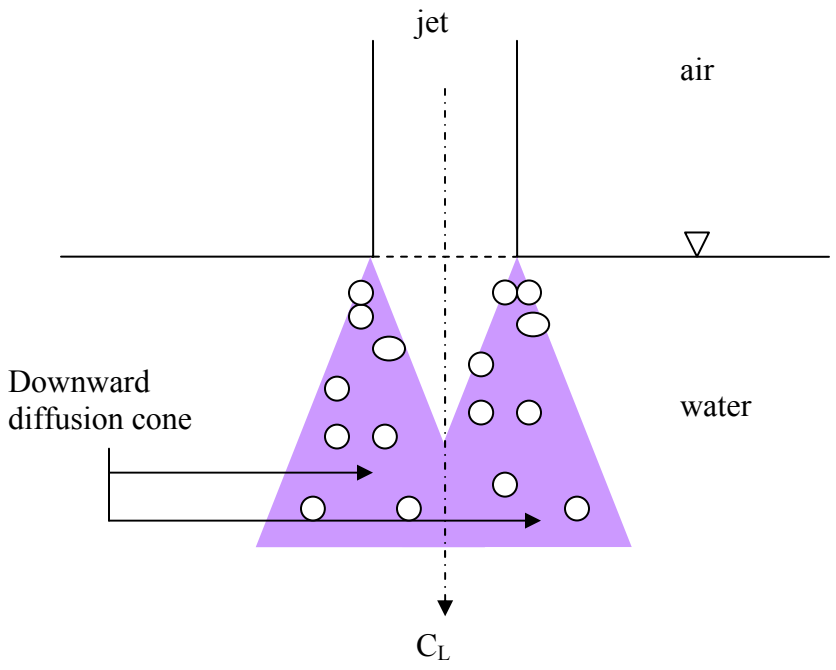

Figure 5-37: Chanson's observation of the air bubble downward diffusion cone from a plunging jet. [24] *p.62*.

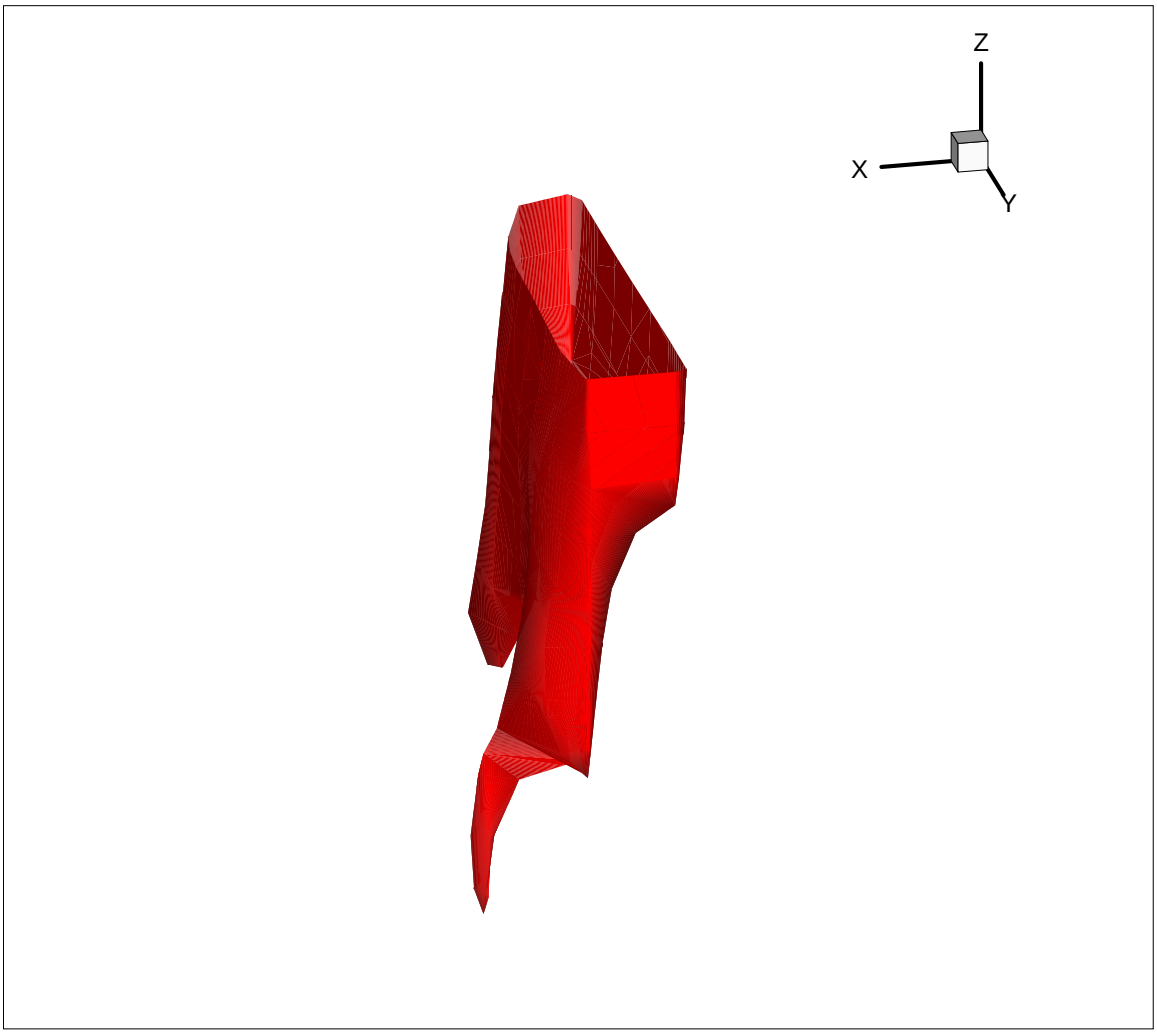

Figure 5-38: Magnification of the modeled *5%* air bubble concentration from Figure 5-32.

At the impact section of the jet, the air concentration was dragged down towards the stagnation point due to the momentum of the jet. Along the way, turbulent fluctuations dispersed the air bubble to the surrounding fluid volumes. At the stagnation point, the strong vertical velocity was converted into horizontal velocities bringing along some air bubbles in the horizontal directions. Since less vertical velocity was present after the jet impact section, the air bubbles had the opportunity to slowly rise to the surface. Therefore the bubble concentration dropped as the distance to the jet impact section increased. However, with the addition of a pump the physics and behavior of the hydrodynamics and air bubbles migration will differ. Air bubbles might not have the opportunity to rise to the surface before entering an operating pump.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

This research resulted in a Large Eddy Simulation finite volume model suited for use in industrial pump station modeling. The model was capable of handling free surface deformation and hydrodynamic pressure variations within a domain. In addition, two different turbulent models namely the classical Smagorinsky SGS model and the dynamic SGS model have been successfully implemented. The subgrid scale models enhanced the capability of the numerical code to simulate problems of different natures.

Several verification simulations were presented for the hydrodynamics and turbulent models. Testing for the hydrodynamics includes laminar and turbulent flow cases with problems of different physics and boundary conditions. For laminar flow cases, the numerical results are compared with analytic solution, while turbulent flow cases were compared with published literature.

In addition to the normal application of a Navier-Stokes solver, the proposed air bubble transport model was added to the code. Modeling of the air bubble dynamics involved two important considerations. The first being the influence of the air bubble towards the LES momentum equations and the second being the influence of the air bubble movement towards the scalar transport equation. A source term was added to the LES momentum equation in the vertical direction to consider the drag force exerted by the air bubbles on the fluid. The scalar transport equation was changed to consider the rise of the air bubble. The rise speed of the air bubble was added to the vertical hydrodynamic velocity to obtain an effective velocity for the air bubble transport. These modifications are general, simple and can be easily implemented to existing Navier-Stokes solvers used in the industry. To verify the air bubble modeling, simulations were carried out to duplicate the experimental data provided in literature.

Boundary conditions from an overfall plunging jet were added as part of the modeling for a pump station. This boundary consisted of the location, angle, magnitude of velocities and the amount of air bubble entering the domain. Upon knowing the drop height and the flow rate at the overfall, the location of the impinging section can be

calculated via particle dynamics. The angle of the impinging jet was calculated from empirical equation, and the angle was used to separate the horizontal and vertical velocity components. An empirical equation for air entrainment quantification was used.

Simple modifications to the code could be made to change the location and sources of air bubbles in order to simulate other engineering problems. Below is a list summarizing the achievements of this work:

1. Proposed and introduced the bubble transport model and implemented the model in a channel flow scenario. The air bubble transport model has been compared with experimental result from previous literature and reasonable agreements near the core of the air bubble concentration were obtained. The results obtained suggested excellent potential to extend its application to problems of a larger scale.
2. Simulated a plunging jet section with air bubbles. The code was used to solve for a hypothetical scenario of a jet impact into a body of still water. Results showed the dynamics of both the fluid and bubbles in the channel setup.

While this work is promising, it opens up a lot more opportunity for future research. Follow up work may include but are not limited to the following list:

1. Experimental work to be conducted for further verification of the proposed model. As suggested earlier in the literature review, there is shortage of data on air bubble concentration movements and migration with the plunging jets. Although the movement of air bubbles may be different due to varying factors (*i.e.* geometry, jet impact angle, flow rate, *etc.*), it is very useful to design and conduct experiments of a simple scenario as a benchmark test case for numerical work. A good experimental setup should consist of a constant head channel or tank with an overfall jet inflow, flow visualization device and air bubble quantification. This experiment should be designed to collect data of the flow rate, drop height, tailwater depth, impact velocities and angle, statistics of bubble size distribution at jet impact section, bubble concentration at different cross sections of the channel and details of the flow pattern in the channel. The objective of this experiment will be to collect air bubble concentration data, flow pattern in a

simple setup as well as to determine the average diameter and concentration of air bubbles at the jet impact section.

2. Another suggestion for future work is to experimentally determine a threshold concentration that could cause significant loss in pump efficiency. A clear allowable limit for volumetric air concentration is needed in order to investigate the effectiveness of any flow configurations or pump placements in a pumping station. Thus far, little information was found regarding the permissible limit of air bubble concentration. Knowing a maximum threshold value for air bubbles concentrations will supply a clear guideline for experimental and numerical pump station modeling to investigate and mitigate bubble interference with a pump.
3. The addition of pump in the system to simulate the actual scenario of a pump station. Pumps can be added into the system by including sink terms for the fluid mass in the domain. For a simplified version of a pump station, a sink near the bottom of the pump station can be first modeled before proceeding to a submersible pump study.
4. Inclusion of different groupings of bubble diameter sizes. Since one diameter is insufficient to provide enough detail for pump station modeling, therefore there is a need to model groups of air bubbles of different sizes. The deficiencies encountered in section 5.7 clearly demonstrated that groups of air bubbles of different sizes maybe needed to produce satisfactory prediction of the concentrations. At a minimum, three different diameters of air bubbles were needed where two to represents each end of the spectrum (maximum and minimum) and one for the average diameter size. Having the three bubble diameter sizes will at the very least give a better approximation of the near wall concentration and provide the information on the effects of the disturbance causes by the larger air bubbles towards the flow domain.
5. Improving simulation time and requirements with parallelization. One avenue worth investigating is the potential of parallel computing, suggested by Dr. Yost and Dr. McDonough. The Pressure Poisson Equation is known to be the bottleneck for serial computations. The preconditioned biconjugate gradient method was used in this study to obtain the solution for the Pressure Poisson

Equation. The preconditioned biconjugate gradient is not well constructed for parallelization. Therefore, other solvers which can be easily parallelized may be a better choice. The parallelization of the code will allow larger problems to be solved in less time and reduces the memory requirements, *e.g.*, in automotive industry, work done by Bauer *et al.* [6] with as much as *9* million cells.

6. To enhance the feature of this code, clustering of the vertical grids is recommended, *e.g.*, the use of the σ-coordinate system. In this work, the author found that uniform vertical grid distribution was not the most efficient way to solve problems pertaining to turbulence, especially when simulating shear bounded flows. In some sense, using uniform vertical grids removed some conceptual advantages offered by the LES formulation. Future work can be done to redistribute the vertical grid by clustering more grids near the wall boundaries. Redistributing the vertical grid helps improve computation time, as well as provide sufficient resolution to resolve turbulence near the walls without having to discretize the grids to a DNS scale resolution for the entire domain.
7. In application of boundary conditions and turbulence modeling, the application of the log law of the wall greatly reduced the grid resolution needed, but came with the expense of some information being truncated from the physical boundary. Turbulent fluctuations near the boundary could not be accurately characterized by the log law of the wall. The use of nested grids seems to be a reasonable choice to overcome this problem. However, having nested grids in the system requires additional grids, which increases computational time. A relationship to establish the missing information from the truncation of the physical boundary is needed to correctly model the turbulence. Perhaps, an empirical relationship could be arrived at for the fluctuating components as well. In addition, at the free surface boundary, the behavior of turbulence differs from that of a typical grid. Turbulence at the free surface becomes anisotropic and thus requires more attention to the subgrid scale model formulation. This particular suggestion is not critical for macroscopic pump station simulation, however, it may help in future studies of near pump flow studies.

8. Simulation of a planar jet hydrodynamics. The planar jet was not simulated in this work. The simulation of the jet require consideration multiple deforming boundaries. At the present time, the code used in this work was unable to handle such phenomena.

APPENDIX A

Transformation of the Navier-Stokes equations to generalized coordinate

The governing Navier-Stokes momentum balance equations were transformed in this work. One of the reason for the transformation was that the Navier-Stokes equations can be solved in a uniformly spaced computational grid with unit volume. This feature will prove to be advantageous in terms of solving problem with complicated geometries. The basics of the transformation will be shown in this section.

To begin the transformation, let ξ, η and ζ be the variables of the transformed equation. The three respective variables are variables of the generalized coordinate system and will be used throughout the discussion of this work. The goal of the transformation is to enable computation to be carried out in uniformly spaced computation grids, however, the physical grid may or may not be uniformly spaced. The transformation begins by first defining the relationship of the variables in the computational domain and the physical domain.

$$\xi = f1(x, y, z) = \xi(x, y, z)$$
$$\eta = f2(x, y, z) = \eta(x, y, z)$$
$$\zeta = f3(x, y, z) = \zeta(x, y, z)$$

Likewise, the mapping from the computational domain to the physical domain is defined as

$$x = g1(\xi, \eta, \zeta) = x(\xi, \eta, \zeta)$$
$$y = g2(\xi, \eta, \zeta) = y(\xi, \eta, \zeta)$$
$$z = g3(\xi, \eta, \zeta) = z(\xi, \eta, \zeta)$$

f(x,y,z) and *g(x,y,z)* are linear mapping functions from the physical domain to computational domain and vice versa. Making use of the relationship shown above, the partial derivatives were obtained by applying the chain rule of differentiation [110]

$$\frac{\partial}{\partial x} = \frac{\partial}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial}{\partial \eta}\frac{\partial \eta}{\partial x} + \frac{\partial}{\partial \zeta}\frac{\partial \zeta}{\partial x} \tag{A.1}$$

$$\frac{\partial}{\partial y}=\frac{\partial}{\partial \xi}\frac{\partial \xi}{\partial y}+\frac{\partial}{\partial \eta}\frac{\partial \eta}{\partial y}+\frac{\partial}{\partial \zeta}\frac{\partial \zeta}{\partial y} \tag{A.2}$$

$$\frac{\partial}{\partial z}=\frac{\partial}{\partial \xi}\frac{\partial \xi}{\partial z}+\frac{\partial}{\partial \eta}\frac{\partial \eta}{\partial z}+\frac{\partial}{\partial \zeta}\frac{\partial \zeta}{\partial z} \tag{A.3}$$

For simplicity, a more convenient set of notations were used and summarized as follows

$$\begin{aligned}
&\frac{\partial \xi}{\partial x}=\xi_x, \quad \frac{\partial \xi}{\partial y}=\xi_y, \quad \frac{\partial \xi}{\partial z}=\xi_z \\
&\frac{\partial \eta}{\partial x}=\eta_x, \quad \frac{\partial \eta}{\partial y}=\eta_y, \quad \frac{\partial \eta}{\partial z}=\eta_z \\
&\frac{\partial \zeta}{\partial x}=\zeta_x, \quad \frac{\partial \zeta}{\partial y}=\zeta_y, \quad \frac{\partial \zeta}{\partial z}=\zeta_z
\end{aligned} \tag{A.4}$$

These terms are also known as the grid metric terms.

Rewriting the equations,

$$\begin{aligned}
\frac{\partial}{\partial x}&=\xi_x\frac{\partial}{\partial \xi}+\eta_x\frac{\partial}{\partial \eta}+\zeta_x\frac{\partial}{\partial \zeta} \\
\frac{\partial}{\partial y}&=\xi_y\frac{\partial}{\partial \xi}+\eta_y\frac{\partial}{\partial \eta}+\zeta_y\frac{\partial}{\partial \zeta} \\
\frac{\partial}{\partial z}&=\xi_z\frac{\partial}{\partial \xi}+\eta_z\frac{\partial}{\partial \eta}+\zeta_z\frac{\partial}{\partial \zeta}
\end{aligned} \tag{A.5}$$

These equations are tidier and easily understood. This form of partial derivatives will be the basis of our transformation. In matrix form, the equation is hence

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} = \begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{bmatrix} \tag{A.6}$$

Similarly, the reverse of the relationship can be written as

$$\begin{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \tag{A.7}$$

Notice that the transformation is the exact inverse of each other. This means that we can simply perform a simple matrix multiplication such as

$$\begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} \begin{bmatrix} \dfrac{\partial}{\partial \xi} \\ \dfrac{\partial}{\partial \eta} \\ \dfrac{\partial}{\partial \zeta} \end{bmatrix} = \begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} \begin{bmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{bmatrix} \begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial z} \end{bmatrix} \tag{A.8}$$

By definition, the left hand side of the equation will become simply the derivatives of the physical domain.

$$\begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial z} \end{bmatrix} = \begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} \begin{bmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{bmatrix} \begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial z} \end{bmatrix} \tag{A.9}$$

From the matrices above, the grid metric matrices have to be the inverse of each other for the equivalency to hold.

$$\begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{bmatrix}^{-1} \tag{A.10}$$

Knowing that, the relationship of the physical domain and the computational domain can be established. Performing the inverse operation, the determinant of the right hand side matrix, also known as inverse of Jacobian transformation can now be defined as *J*.

$$\begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{bmatrix}^{-1} = J \begin{bmatrix} y_\eta z_\zeta - y_\zeta z_\eta & y_\zeta z_\xi - y_\xi z_\zeta & y_\xi z_\eta - y_\eta z_\xi \\ x_\zeta z_\eta - x_\eta z_\zeta & x_\xi z_\zeta - x_\zeta z_\xi & x_\eta z_\xi - x_\xi z_\eta \\ x_\eta y_\zeta - x_\zeta y_\eta & x_\zeta y_\xi - x_\xi y_\zeta & x_\xi y_\eta - x_\eta y_\xi \end{bmatrix} \tag{A.11}$$

where,

$$J = \frac{1}{x_\xi \left(y_\eta z_\zeta - y_\zeta z_\eta \right) - x_\eta \left(y_\xi z_\zeta - y_\zeta z_\xi \right) + x_\zeta \left(y_\xi z_\eta - y_\eta z_\xi \right)}$$

Substituting this relationship in the Navier-Stokes equation, will give the transformation required of the Cartesian to a generalized coordinate system.

Since the grid was transformed, the velocity vectors need to also be altered in such a way that the computational velocity will be at the location and direction desired for direct calculations. The velocities transformation is similar and more straightforward. The chain rule was employed to obtain

$$\begin{aligned} u &= u_\xi \frac{\partial x}{\partial \xi} + u_\eta \frac{\partial x}{\partial \eta} + u_\zeta \frac{\partial x}{\partial \zeta} \\ v &= u_\xi \frac{\partial y}{\partial \xi} + u_\eta \frac{\partial y}{\partial \eta} + u_\zeta \frac{\partial y}{\partial \zeta} \\ w &= u_\xi \frac{\partial z}{\partial \xi} + u_\eta \frac{\partial z}{\partial \eta} + u_\zeta \frac{\partial z}{\partial \zeta} \end{aligned} \tag{A.12}$$

Likewise,

$$\begin{aligned} u_\xi &= u\xi_x + v\xi_y + w\xi_z \\ u_\eta &= u\eta_x + v\eta_y + w\eta_z \\ u_\zeta &= u\zeta_x + v\zeta_y + w\zeta_z \end{aligned} \tag{A.13}$$

For completeness, the matrix form of equation (A.12) and equation (A.13) are

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix} \begin{bmatrix} u_\xi \\ u_\eta \\ u_\zeta \end{bmatrix} \tag{A.14}$$

and,

$$\begin{bmatrix} u_\xi \\ u_\eta \\ u_\zeta \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{A.15}$$

This will complete the generalized coordinate transformation section. To obtain the transformed Navier-Stokes equations, substitution of terms derived in this section into the Navier-Stokes equations will produce the set of equation used for this work. However, the procedure to transform the Navier-Stokes momentum equations is very tedious and time consuming. As mentioned earlier, it will be very helpful to gain some mathematical background and ideas from literature to successfully perform the transformation .

Transformation of the continuity equation to generalized coordinate

A simple example of transformation for the continuity equation will be shown. In Cartesian coordinate system, the equation for continuity was previously shown in equation (2.1.1) and is rewritten here

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0$$

Applying the relationship shown in equation (A.1) through equation (A.3), results in

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=\left(\frac{\partial u}{\partial \xi}\xi_x+\frac{\partial u}{\partial \eta}\eta_x+\frac{\partial u}{\partial \zeta}\zeta_x\right)+\left(\frac{\partial v}{\partial \xi}\xi_y+\frac{\partial v}{\partial \eta}\eta_y+\frac{\partial v}{\partial \zeta}\zeta_y\right)+\left(\frac{\partial w}{\partial \xi}\xi_z+\frac{\partial w}{\partial \eta}\eta_z+\frac{\partial w}{\partial \zeta}\zeta_z\right)$$

Rearranging the right hand side of the equations yields

$$=\left(\frac{\partial u}{\partial \xi}\xi_x+\frac{\partial v}{\partial \xi}\xi_y+\frac{\partial w}{\partial \xi}\xi_z\right)+\left(\frac{\partial u}{\partial \eta}\eta_x+\frac{\partial v}{\partial \eta}\eta_y+\frac{\partial w}{\partial \eta}\eta_z\right)+\left(\frac{\partial u}{\partial \zeta}\zeta_x+\frac{\partial v}{\partial \zeta}\zeta_y+\frac{\partial w}{\partial \zeta}\zeta_z\right)$$

Further manipulation yields

$$=\frac{\partial}{\partial \xi}\left(u\xi_x+v\xi_y+w\xi_z\right)+\frac{\partial}{\partial \eta}\left(u\eta_x+v\eta_y+w\eta_z\right)+\frac{\partial}{\partial \zeta}\left(u\zeta_x+v\zeta_y+w\zeta_z\right)$$

In using the relationship shown in equation (A.13), the conversion of the Cartesian velocities to contravariant velocities will produce the final form of the transformed continuity equation and is shown as

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=\frac{\partial u_\xi}{\partial \xi}+\frac{\partial u_\eta}{\partial \eta}+\frac{\partial u_\zeta}{\partial \zeta}=0$$

For further details of the derivation and coordinate transformation, readers are referred to Anderson *et al.*'s book entitled Computational Fluid Mechanics and Heat Transfer [2].

The calculation of the each mapping term can be simplified by choosing $\Delta\xi=\Delta\eta=\Delta\zeta=1$, hence, when evaluating the flux terms, one can simply use any preferred spatial discretization scheme to evaluate x_ξ, x_η, x_ζ, y_ξ, y_η y_ζ, z_ξ, z_η and z_ζ. For completeness, a simple example, using a typical finite difference discretization, with a central differencing scheme to obtain x_ξ is

$$x_\xi=\frac{x_{j+1,k,l}-x_{j-1,k,l}}{2(\Delta\xi)}=\frac{x_{j+1,k,l}-x_{j-1,k,l}}{2}$$

The same operation can be done to complete the mapping. To calculate the grid metric terms, x_ξ, x_η, x_ζ, y_ξ, $y_\eta y_\zeta$, z_ξ, z_η and z_ζ will be first calculated and used to evaluate J, the Jacobian of transformation. The matrix form previously shown as (A.11) will be used to obtain the grid metrics.

Central differencing at a cell face and the free surface equation

Before arriving at the computational form for the free surface and the pressure equation, the equation involves central differencing the gradient terms at a cell face. This procedure is not very much different than the normal central difference formulation. The first order central difference formula for a variable *f(x)* in a cell is

$$\frac{\partial f(x)}{\partial x} = \frac{f(x+h) - f(x-h)}{2} \tag{A.16}$$

A typical central differencing scheme assumes that all variables are arranged the same way in all cells in the domain. Therefore, the procedure to centrally discretize the first order gradient terms is straightforward. In this work, since the vectors are located at the cell faces while the scalar variables are located at the center of a cell, the discretization takes on an additional step. For simplicity, an example was shown on the discretization of the free surface term.

From the free surface section, the equation

$$h_{j,k}{}^{n+1} - (\theta\Delta t)^2 \left\{ \left(H_\xi{}^{n+1} D \right)_{j+1/2,k} - \left(H_\xi{}^{n+1} D \right)_{j-1/2,k} + \left(H_\eta{}^{n+1} D \right)_{j,k+1/2} - \left(H_\eta{}^{n+1} D \right)_{j,k-1/2} \right\} =$$

$$h_{j,k}{}^{n} - (1-\theta)\Delta t \sum_{l=1}^{nl} \left[\left(u_\xi{}^{n} \Delta z \right)_{j+1/2,k,l} - \left(u_\xi{}^{n} \Delta z \right)_{j-1/2,k,l} + \left(u_\eta{}^{n} \Delta z \right)_{j,k+1/2,l} - \left(u_\eta{}^{n} \Delta z \right)_{j,k-1/2,l} \right] -$$

$$\theta\Delta t \sum_{l=1}^{nl} \left[\begin{array}{l} \left(\left(u_\xi{}^{n} - \Delta t \left(F_\xi - s_\xi \right)^{n+1/2} - (1-\theta)\Delta t H_\xi{}^{n} \right) \Delta z \right)_{j+1/2,k,l} - \\ \left(\left(u_\xi{}^{n} - \Delta t \left(F_\xi - s_\xi \right)^{n+1/2} - (1-\theta)\Delta t H_\xi{}^{n} \right) \Delta z \right)_{j-1/2,k,k} + \\ \left(\left(u_\eta{}^{n} - \Delta t \left(F_\eta - s_\eta \right)^{n+1/2} - (1-\theta)\Delta t H_\eta{}^{n} \right) \Delta z \right)_{j,k+1/2,l} - \\ \left(\left(u_\eta{}^{n} - \Delta t \left(F_\eta - s_\eta \right)^{n+1/2} - (1-\theta)\Delta t H_\eta{}^{n} \right) \Delta z \right)_{j,k-1/2,l} \end{array} \right]$$

contains the gradients H that has to be centrally discretized. For discussion sake, only one term, $\left(H_\xi{}^{n+1}D\right)_{j+1/2,k}$ will be discretized. Previously, after the coordinate transformation, the term becomes

$$\left(H_\xi{}^{n+1}D\right)_{j+1/2,k}=\left[g\left(l_{\xi\xi}\frac{\partial h}{\partial\xi}+l_{\xi\eta}\frac{\partial h}{\partial\eta}\right)D\right]_{j+1/2,k} \tag{A.17}$$

$$\left(H_\xi{}^{n+1}D\right)_{j+1/2,k}=g\left(l_{\xi\xi}\frac{\partial h_{j+1/2,k}}{\partial\xi}+l_{\xi\eta}\frac{\partial h_{j+1/2,k}}{\partial\eta}\right)D_{j+1/2,k} \tag{A.18}$$

$$\begin{aligned}\frac{\partial h_{j+1/2,k}}{\partial\xi}&=\frac{h_{j+1,k}-h_{j,k}}{2}\\ \frac{\partial h_{j+1/2,k}}{\partial\eta}&=\frac{h_{j+1/2,k+1/2}-h_{j+1/2,k-1/2}}{2}\end{aligned} \tag{A.19}$$

The first term is a straightforward central discretization because the result of the discretization is a function of h at cell center. The second term produces h at a cell face. To treat the inconsistent location of h, simple interpolation will be performed, utilizing the neighboring cell center values to produce the same effect as that at the cell face values.

$$h_{j+1/2,k+1/2}=\frac{h_{j+1,k+1}+h_{j,k+1}}{2} \tag{A.20}$$

Similarly,

$$h_{j+1/2,k+1/2}=\frac{h_{j+1,k-1}+h_{j,k-1}}{2} \tag{A.21}$$

Substituting the equation back, one obtains

$$\frac{\partial h_{j+1/2,k}}{\partial\eta}=\frac{\dfrac{h_{j+1,k+1}+h_{j,k+1}}{2}-\dfrac{h_{j+1,k-1}+h_{j,k-1}}{2}}{2}=\frac{h_{j+1,k+1}+h_{j,k+1}-h_{j+1,k-1}-h_{j,k-1}}{4} \tag{A.22}$$

Resulting in

$$\left(H_\xi{}^{n+1}D\right)_{j+1/2,k}=g\left(l_{\xi\xi}\frac{h_{j+1,k}-h_{j,k}}{2}+l_{\xi\eta}\frac{h_{j+1,k+1}+h_{j,k+1}-h_{j+1,k-1}-h_{j,k-1}}{4}\right)D_{j+1/2,k} \tag{A.23}$$

The discretization is complete for the $\left(H_{\xi}^{n+1}D\right)_{j+1/2,k}$ term and all the variable h is now consistently located in cell center. Similar discretization was carried out for all remaining terms to produce the simplified form of the free surface equation.

The simple *2*-dimensional cells in Figure A-1 will illustrates the discretization process that involves the cell center points

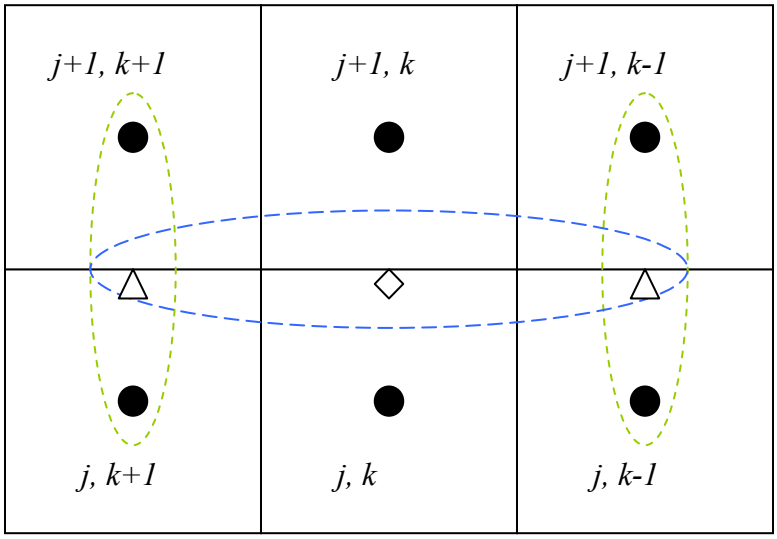

Figure A-1: Discretization stencil and interpolation to obtain cell face values

To obtain the gradient at the cell face shown in the figure as the diamond, the discretization in the η direction requires the h values at the location indicated by the triangles. The nearest cell centered value to the triangles will be interpolated to obtain the values of the triangles. Once the values of the triangles are known, the typical central differencing takes place to obtain the value at the diamonds.

APPENDIX B

As a summary of the computer code used, the pseudo code which outlines the structure of the code is as follows:

- Acquire input data and initial conditions, boundary type, u,v,w,h, *etc.*
- Read in physical coordinate points and connectivity
- Calculate grid metric terms from physical coordinate
- Transform Cartesian u, v, w to computational coordinate u_ξ ,u_η,u_ζ
- Determine the location that the free overfall at the surface of the domain
- Calculate/obtain initial bubble concentration, ϕ^n
- input initial condition, u_i^n, h^n

for t = 1 to nt

- calculate predictor velocities, $u_i^{n+1/2}$ from old time level u_i^n, s_i^n, h^n and p^n
- calculate source terms, $s_i^{n+1/2}$, $F_i^{n+1/2}$, $s_i^{n+1/2}$ from $u_i^{n+1/2}$
- calculate approximate velocities $u_i{}^{*n+1}$, from $F_i^{n+1/2}$, $s_i^{n+1/2}$ and u_i^n, $p = 0$
- approximate free surface elevation, h^{n+1} from h^n, u_i^n, $u_i{}^{*n+1}$
- calculate approximate velocities u^{**n+1} for pressure calculation, from p^n, u_i^n, $F_i^{n+1/2}$ and $s_i^{n+1/2}$
- solve for dynamic pressures, p^{n+1}, knowing u^{**n+1}
- calculate new velocities for $n+1$ time level, u_i^{n+1}
- Calculate source term for scalar transport equation, s_ϕ^n from u_i^n, ϕ^n
- calculate predictor scalar transport, $\phi^{n+1/2}$
- Calculate source term for scalar transport equation, s_ϕ^{n+1} from $u_i^{n+1/2}$, $\phi^{n+1/2}$
- Compute corrector step to find ϕ^{n+1}
- if $t = tplot$
    - write output - u_ξ , u_η , u_ζ, p, h, ϕ, ν_t, etc.

  end if
- update $u_i^n=u_i^{n+1}$, $h^n=h^{n+1}$, $p^n=p^{n+1}$, $\phi^n=\phi^{n+1}$

return
end loop

- post process the output

The pseudo code is a crude summary of the simulation steps. Implementation of each step requires more further details and should be done with care.

REFERENCES

[1] Afzal, N., *Power law and log law velocity profiles in turbulent boundary-layer flow: Equivalent relations at large Reynolds numbers.* Acta Mechanica, 2001. **151**(3-4): p. 195-216.

[2] Anderson, D.A., Tannehill, J. C., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*. 1984, New York: Hemisphere Publishing Corp. 252-255.

[3] Ansar, M., Nakato, T., and Constantinescu, S.G., *Numerical simulation of inviscid three-dimensional flows at single and dual pump intakes.* Journal of Hydraulic Research, 2002. **40**(4): p. 461-470.

[4] Apsley, D., *CFD Calculation of Turbulent Flow with Arbitrary Wall Roughness.* Flow, Turbulence and Combustion, 2007. **78**(2): p. 153-175.

[5] Babajimopoulos, C. and Bedford, K.W., *Formulating lake models which preserve specral statistics* ASCE Journal of Hydraulic Division, 1980. **106**(1): p. 1-19.

[6] Bauer, W., Reister, H., Ross, F., and Robinson, D., *Parallel Computing for CFD Applications in the Automotive Industry - First Experiences*. HPCN Europe 1996: Proceedings of the International Conference and Exhibition on High-Performance Computing and Networking. 1996, London, UK: Springer-Verlag.

[7] Bedford, K.W. *Turbulence charateristics of wind-driven shallow lake models*. in *Proceedings of the Symposium - International Association for Hydraulic Research: Refined Modelling of Flows*. 1982. Paris, France: Presses de l'Ecole Natl des Ponts et Chaussees, Paris, Fr.

[8] Bin, A.K., *Gas Entrainment by plunging liquid jets.* Chemical Engineering science, 1993. **48**(21): p. 3585-3630.

[9] Blumberg, A.F. and Mellor, G.F., *A description of a three-dimensional coastal ocean circulation model*, in *Three Dimensional Coastal Ocean Models*, N.S. Heaps, Editor. 1987: Washington D.C. p. 1-16.

[10] Bohrer, J.G. and Abt, S.R., *Plunge Pool Velocity Decay of Rectangular Free Falling Jets. Dam Foundation Erosion. Phase II . Clear Water Experiments, 1:3 Scale Model Facility*. 1996, Bureau of Reclamation: Denver.

[11] Bohrer, J.G., Abt, S.R., and Wittler, R.J., *Predicting plunge pool velocity decay of free falling, rectangular jets.* Journal of Hydraulic Engineering, ASCE, 1998. **124**(HY10): p. 1043-1048.

[12] Borghei, S.M. and Etminani, A.A. *Effect of plunge pool width on hydrodynamic pressure due to vertical jet*. in *XXIX IAHR Congress*. 2001. Beijing, China.

[13] Bozzano, G. and Dente, M., *Shape and terminal velocity of single bubble motion:A novel approach.* Computers and Chemical Engineering, 2001. **25**: p. 571-576.

[14] Bradford, S. and Katapodes, N.D., *Monotonicity Preserving Model for Free Surface Flow with Sharp Density Fronts*, in *Estuarine and Coastal Modeling*. 1999: New Orleans.

[15] Bradford, S.F. and Sanders, B.F., *Modeling flows with moving boundaries due to flooding, recession and wave run-up*, in *Proceedings of the Seventh International Conference on Estuarine and Coastal Modeling*, ASCE, Editor. 2001: St. Petersburg, Florida. p. 695-708.

[16] Brattberg, T. and Chanson, H., *Air entrapment and air bubble dispersion at two-dimensional plunging jets.* Chemical engineering science, 1998. **53**(24): p. 4113-4127.

[17] Cabot, W. and Moin, P., *Approximate Wall Boundary Conditions in the Large Eddy Simulation of High Reynolds NUmber Flow.* Flow, Turbulence and Combustion, 1999. **63**: p. 269-291.

[18] Canepa, S. and Hager, W.H., *Effect of jet air content on plunge pool scour.* Journal of Hydraulic Engineering ASCE, 2003. **129**(5): p. 358-365.

[19] Chamani, M.R. and Beirami, M., *Flow charateristics at drop.* Journal of Hydraulic Engineering, ASCE, 2002. **128**(8): p. 788-791.

[20] Chan, R.K.C. and Street, R.L., *A computer study of finite-amplitude water waves.* Journal of Computational Physics, 1970. **6**: p. 68-94.

[21] Chanson, H., *Hydraulics of Nappe Flow Regime above Stepped Chutes and Spillways.* Australian Civil/Structural Engineering Transactions, 1994. **CE36**(1): p. 69-76.

[22] Chanson, H., *Air entrainment in two-dimensional turbulent shear flows with partially developed inflow conditions.* International Journal of Multiphase Flow, 1995. **21**(6): p. 1107-1121.

[23] Chanson, H., *Air bubble diffusion in supercritical open channel flows*, in *Twelfth Australasian Fluid Mechanics Conference*. 1995: The University of Sydney, Australia.

[24] Chanson, H., *Air bubble entrainment in free surface turbulent shear flows*. 1996, San Diego: Academic Press, Inc.

[25] Chanson, H., *Environmental Hydraulics of Open Channel Flows*. 2004, Oxford, UK: Elsevier Butterworth-Heinemann. 430.

[26] Chanson, H. and Brattberg, T. *Air entrainment by two-dimensional plunging jets: The impingement region and the very-near flow field*. in *Proceeding of FEDSM'98, 1998 ASME Fluids Engineering Division Summer Meeting*. 1998. Washington, DC.

[27] Char, M.I. and Hsu, Y.H., *Computation of buoyancy-driven flow in an eccentric centrifugal annulus with a non-orthogonal collocated.* International Journal For Numerical Methods In Fluids, 1998. **26**: p. 323–343

[28] Chen, C. and Fan, L., *Discrete simulation of gas-liquid bubble columns and gas-liquid-solid fluidized beds.* AIChE Journal, 2004. **50**(2): p. 288-301.

[29] Chen, J., Meneveau, C., and Katz, J., *Scale interactions of turbulence subjected to a straining–relaxation–destraining cycle.* Journal of Fluid Mechanics, 2006. **562**: p. 123-150.

[30] Chen, P., Dudukovic, M.P., and Sanyal, J., *Three-dimensional simulation of bubble column flows with bubble coalescence and breakup.* AIChE journal 2005. **51**(3): p. 288-301.

[31] Choi, H. and Moin, P., *Effects of Computational time step on numerical solutions of turbulent flow.* Journal of Computational Physics, 1994. **113**: p. 1-4.

[32] Chow, V.T., *Open Channel Hydraulics*. 1959, New York McGraw Hill Book Co., Inc.

[33] Chung, T.J., *Computational fluid dynamics*. illustrated, reprint ed. 2002: Cambridge University Press.

[34] Cienfuegos, R., Barthelemy, E., and Bonneton, P., *A Fourth-Order Compact Finite Volume Scheme for Fully Nonlinear and Weakly Dispersive Boussinesq-type equation. Part II: Boundary Conditions and Validation.* International Journal For numerical methods in Fluids, 2007. **53**: p. 1423-1455.

[35] Clift, R., Grace, J.R., and Weber, M.E., *Bubbles, Drops, and Particles* 1978, New York: Academic Press. 380.

[36] Constantinescu, S.G. and Patel, V.C., *Numerical model for simulation of pump-intake flow and vortices.* Journal of Hydraulic Engineering, ASCE, 1998. **124**(2): p. 123-134.

[37] Constantinescu, S.G. and Patel, V.C., *Role of turbulence model in prediction of pump-bay vortices.* Technical Paper, Journal of Hydraulic Engineering, ASCE, 2000. **126**(5): p. 387-392.

[38] Cummings, P.D. and Chanson, H., *Air entrainment in the developing flow region of plunging jets. 2. Experimental.* Transactions of the ASME, 1997. **119**(3): p. 603-608.

[39] Cummings, P.D. and Chanson, H., *Air entrainment in the developing flow region of plunging jets. 1. Theoretical development.* Transactions of the ASME. Journal of Fluids Engineering, 1997. **119**(3): p. 597-602.

[40] Davidson, L. *Hybrid LES-RANS: Inlet Boundary Conditions*. in *3rd National Conference on Computational Mechanics-MekIT'05*. 2005. Trondheim, Norway: Eds. B. Skallerud and H.I. Anderson.

[41] Davidson, L. and Billson, M., *Hybrid LES-RANS using synthesized turbulence for forcing at the interface*, in *Fourth European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS 2004*. 2004, P. Neittaanmäki, T. Rossi, S. Korotov, E. Oñate, J. Périaux, and D. Knörzer (eds.): Jyväskylä, Finland.

[42] Davidson, L. and Billson, M., *Hybrid LES-RANS using synthesized turbulent fluctuations for forcing in the interface region.* International Journal of Heat and Fluid Flow, 2006. **27**: p. 1028-1042.

[43] Deardorff, J.W., *A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers.* Journal of Fluid Mechanics, 1970. **41**: p. 453.

[44] Deen, N.G., van Sint Annaland, M., and Kuipers, J.A.M., *Multi-scale modeling of dispersed gas–liquid two-phase flow* Chemical Engineering Science, 2004. **59**(8): p. 1853-1861.

[45] di Mare, L., Klein, M., Jones, W.P., and Janicka, J., *Synthetic turbulence inflow conditions for large eddy simulation.* Physics of Fluids, 2006. **18**: p. 025107.1 - 025107.11.

[46] Ding, X., *Large eddy simulation of turbulent transport processes by a least-squares finite element method / by Xu Ding*, in *Department of Chemical and Materials Engineering*. 1999, University of Kentucky: Lexington. p. 133.

[47] Domaradzki, A.J. and Saiki, E.M., *A subgrid-scale model based on the estimation of unresolved scale of turbulence.* Physics of Fluids, 1997. **9**(7): p. 2148-2164.

[48] Dong, Z., Gao, S., and Fryrear, D.W., *Drag coefficients, rougness length and zero-plane displacement height as disturbed by artificial standing vegetation.* Journal of Arid Environments, 2001. **49**: p. 485-505.

[49] El Hammoumi, M., Achard, J.L., and Davoust, L., *Measurements of air entrainment by vertical plunging liquid jets.* Experiments in fluids, 2002. **32**: p. 624-638.

[50] Ervine, D.A. and Elsawy, E.M. *The effect of falling nappe on river aeration*. in *Proc 16th IAHR Congress*. 1975. Sao Paulo, Brazil.

[51] Ervine, D.A. and Falvey, H.T. *Behaviour of turbulent water jets in the atmosphere and in plunge pools*. in *Proc. Inst. Civ. Engrs., Part 2*. 1987.

[52] Ervine, D.A., Falvey, H.T., and Withers, W., *Pressure fluctuations on plunge pool floors.* Journal of Hydraulic Research, 1997. **35**(2): p. 157-279.

[53] Ferziger, J.H., *Large eddy simulation*, in *Simulation and Modeling of Turbulent Flows*, T.B. Gatski, Hussaini, M.Y., & Lumley, J.L., Editor. 1996, Oxford University Press: United States.

[54] Ferziger, J.H. and Peric, M., *Computational Methods for Fluid Dynamics*. 2001, Berlin: Springer; 3rd rev. ed. edition (December 12, 2001).

[55] Finnemore, E.J. and Franzini, J.B., *Fluid mechanics with engineering applications*. tenth edition ed. 2002, Boston, Massachusetts: McGraw-Hill.

[56] Friedrich, R., Huttl., T.J., Manhart, M., and Wagner, C., *Direct numerical simulation of incompressible turbulent flows.* Computers and Fluids, 2001. **30**: p. 555-579.

[57] Germano, M., Piomelli, U., Moin, P., and W.H., C., *A dynamic subgrid-scale eddy viscosity model.* Physics of Fluids A, 1991. **3**(7): p. 1760-1765.

[58] Ghosal, S., Lund., T.S., Moin, P., and Akselvoll, K., *A dynamic localization model for Large-Eddy simulation of turbulent flows.* Journal of fluid mechanic, 1995. **286**: p. 229-255.

[59] Gollub, J., *Research in fluid dynamics: meeting national needs*, in *A report of the U.S. National Committee on Theoretical and Applied Mechanics*. 2006.

[60] Gresho, P.M., *Incompressible fluid dynamics:some fundamental formulation issues.* Annual Review of Fluid Mechanics, 1991. **23**: p. 413-454.

[61] Gunther, A., Papavassilliou, D.V., Warholic, M.D., and Hanratty, T.J., *Turbulent flow in a channel at a low Reynolds number.* Experiment in Fluids, 1998. **25**(5/6): p. 503-511.

[62] Haberman, W.M. and Morton, R.K., *Experimental Study of Bubbles Moving in Liquids.* Trans. Amer. Sot. Civil Eng., 1956. **121**: p. 227-252.

[63] Hadjerioua, B., Laursen, M.E., Peterson, S.M., and Rizk, T.A. *Air entrainment and bubble behavior in plunge pools*. in *National Conference on Hydraulic Engineering Proceedings of the 1994 Conference* 1994. Buffalo, New York.

[64] Hanjalic, K., *Will RANS survive LES? A view of perspectives.* Journal of fluid engineering, 2005. **127**: p. 831-839.

[65] Hanstead, A.R., *Measurement of static liquid holdup at low EOTVOS numbers*, in *Department of Chemical Engineering*. 2000, MeGill University: Montreal. p. 72.

[66] Harlow, F.H. and Besnard, D., *Well-posed two-phase flow equations with turbulence transport.* Letters in Mathematical Physics, 1985. **10**: p. 161-166.

[67] Hinata, S., Kuga, O., and Kobayashi, K., *Diffusion of bubbles in Two-Phase Flow* (The 1st report, on the method for measurement of the diffusivity of the bubble).* Bulletin of the JSME, 1977. **20**(148): p. 1299-1305.

[68] Hinze, J.O., *Turbulence, An introduction to its mechanism and theory*. 1959, New York: McGraw-Hill 586.

[69] Horiuti, K., *Large eddy simulation of turbulent channel flow by one-equation modeling.* Journal of the Physical Society of Japan, 1985. **54**(8): p. 2855-2865.

[70] Hsu, M.H., Chen, C.F., and Teng, W., H., *An arbitrary Lagrangian-Eulerian finite difference method for computations of free surface flows.* Journal of Hydraulic Research, 2000. **39**(4): p. 1 - 11.

[71] Ishii, M. and Zuber, N., *Drag coefficient and relative velocity in bubbly, droplet and particulate flows.* AIChE Journal, 1979. **25**(5): p. 843-855.

[72] Ishikawa, Y., Mzuhara, K., and Ashida, A., *Effect of density of trees on drag exerted on trees in river channels.* Journal of Forest Research, 2000. **5**: p. 271-279.

[73] Jarrin, N., Benhamadouche, S., Laurence, D., and Prosser, R., *A synthetic-eddy method for generating inflow conditions for large-eddy simulations.* International Journal of Heat and Fluid Flow, 2006. **27**: p. 585-593.

[74] Jimenez, J. and Moin, P., *The minimal flow unit in near-wall turbulence.* Journal of Fluid Mechanics, 1991 **225**: p. 213-240.

[75] Kawamura, T. and Kodama, Y., *Numerical simulation method to resolve interactions between bubbles and turbulence.* International Journal of Heat and Fluid Flow, 2002. **23**: p. 627-638.

[76] Kleefsman, K.M.T., Fekken, G., Veldman, A.E.P., Iwanowski, B., and Buchner, B., *A Volume-of-Fluid based simulation method for wave impact problems.* Journal of Computational Physics, 2005. **206** p. 363–393.

[77] Klein, M., Sadiki, A., and Janicka, J., *A digital filter based generation of inflow data for spatially developing direct numerical or large eddy simulations.* Journal of Computational Physics, 2003. **186**: p. 652-665.

[78] Kostaschuk, R., Villard, P., and Best, J., *Measuring velocity and shear stress over dunes with acustic doppler profiler.* Journal of Hydraulic Engineering ASCE, 2004. **130**(9): p. 932-936.

[79] Lee, J.W., Teubner, M.D., Nixon, J.B., and Gill, P.M., *Applications of the artificial compressibility method for turbulent open channel flows.* International Journal For Numerical Methods In Fluids, 2006. **51**: p. 617–633.

[80] Lesieur, M. and Metais, O., *New trends in Large-Eddy simulations of turbulence.* Annual Review of Fluid Mechanics, 1996. **28**: p. 45-82.

[81] Liao, H., Xu, W., Yang, Y., and Wu, C., *Numerical simulation of a 3-D turbulent flows of plunge pool and energy dissipation analysis.* Journal of Hydrodynamics, 1997. **B**(1): p. 1-10.

[82] Lilly, D.K., *A proposed modification of the Germano subgrid-scale closure model.* Physics of Fluids A, 1992. **4**(3): p. 633-635.

[83] Lomax, H., Pulliam, T.H., and Zingg, D.W., *Fundamentals of Computational Fluid Dynamics*. 2001: Springer-Verlag.

[84] Lund, T.S., Wu, X., and Squires, K.D., *Generation of turbulent inflow data for spatially-developing boundary layer simulations.* Journal of Computational Physics, 1998. **140**: p. 233-258.

[85] Majander, P. and Siikonen, T., *Evaluation of Smagorinsky-based subgrid-scale models in a finite-volume computation.* International Journal For numerical methods in Fluids, 2002. **40**(6): p. 735-774.

[86] Mason, P.J., *Effects of air entrainment on plunge pool scour.* Journal of Hydraulic Engineering ASCE, 1989. **115**(3): p. 385-399.

[87] Mason, P.J. and Callen, N.S., *On the magnitude of the subgrid-scale eddy coefficient in large-eddy simulations of turbulent channel flow.* Journal of fluid mechanic, 1986. **162**: p. 439-462.

[88] McDonough, J.M., *Introductory Lecture on Turbulence – Physics, mathematics and modeling*. 2004: Department of Mechanical Engineering and Mathematics, University of Kentucky, Lexington, Kentucky.

[89] Medic, G., Templeton, J.A., and Kalitzin, G., *Wall modeling for LES: What turbulence information is retained?*, in *Conference on turbulence and interactions TI2006*. 2006: Porquerolles, France.

[90] Melo, J.F. *Plunging jets - Pool floor mean dynamic pressure reduction due to aeration*. in *Proc., International Workshop on Rock Scour Due to High-Velocity Jets*. 2002. Lausanne, Switzerland, Anton J. Schleiss and E. Bollaert, eds. Balkema, Rotterdam, The Netherlands.

[91] Mendelson, H.D., *The prediction of bubble terminal velocities from wave theory.* A.I.Ch.E Journal, 1967. **13**: p. 250-253.

[92] Moin, P. and Kim, J., *On the Numerical Solution of Time-Dependant Viscous Incompressible Fluid Flows Involving Solid Boundaries.* Journal of Computational Physics, 1980. **35**: p. 381-392.

[93] Moin, P. and Kim, J., *Numerical investigation of turbulent channel flow.* Journal of Fluid Mechanics, 1982. **118**: p. 341-377.

[94] Moin, P., *Advances in large eddy simulation methodology for complex flows.* International Journal of Heat and Fluid Flow, 2002. **23**: p. 710-720.

[95] Moser, R.D., Kim, J., and Mansour, N.N., *Direct Numerical simulation of turbulent channel flow up to Re=590.* Physics of Fluids, 1999. **11**(4): p. 943-945.

[96] Munson, B.R., Young, D.F., and Okiishi, T.H., *Fundamentals of fluid mechanics*. Third edition ed. 1999, New York, USA: John Wiley and Sons.

[97] Orlanski, I., *A simple boundary condition for unbounded hyperbolic flows.* Journal of Computationsl Physics, 1976. **21**(3): p. 251-269.

[98] Piomelli, U. and Balaras, E., *Wall-Layer Models for Large-Eddy Simulations.* Annual review in fluid mechanics, 2002. **34**: p. 349-374.

[99] Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., *Numerical Recipes in FORTRAN 77: The Art of Scientific Computing*. 2 edition ed. Vol. 1. 1992, New York: Cambridge University Press. 992.

[100] Rajendran, V., Constantinescu, S.G., and Patel, V.C., *Experimental validation of a numerical model of flow in pump-intakes bays.* Journal of Hydraulic Engineering ASCE, 1999. **125**(11): p. 1119-1126.

[101] Ramamurthy, A.S., Qu, J., and Vo, D., *Volume of fluid model for an open channel flow problem.* Canadian Journal of Civil Engineering, 2005. **32**: p. 996-1001.

[102] Roberson, J.A. and Crowe, C.T., *Engineering fluid mechanics*. Sixth edition ed. 1997, New York, USA: John Wiley & Sons.

[103] Scotti, A. and Meneveau, C., *Generalized Smagorinsky model for anisotropic grids.* Physics of Fluids A, 1993. **5**(9): p. 2306-2308.

[104] Scotti, A., Meneveau, C., and Fatica, M., *Dynamic Smagorinsky model on anisotropic grids.* Physics of Fluids, 1997. **9**(6): p. 1856-1858.

[105] Sethian, J.A., *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. 2nd edition ed. 1999: Cambridge University Press.

[106] Shew, W.L., Poncet, S., and Pinton, J.F., *Force measurements on rising bubbles.* Journal of fluid mechanic, 2006. **569**: p. 51-60.

[107] Simonnet, M., Gentric, C., Olmos, E., and Midoux, N., *Experimental determination of the drag coefficient in a swarm of bubbles.* Chemical Engineering science, 2007. **62**(3): p. 858-866

[108] Smolianski, A., Heikki, H., and Pasi, L., *Numerical study of dynamics of single bubbles and bubble swarms.* Applied Mathematical Modelling, 2008. **32**(5): p. 641-659.

[109] Sommerfeld, M., *Bubbly Flows: Analysis, Modelling and Calculation*. Martin Sommerfeld ed. Heat and Mass Transfer, ed. D. Mewes and F. Mayinger. 2004, New York: Springer Verlag. 351.

[110] Stewart, J., *Calculus*. Third edition ed. 1995, California, USA: Brooks/Cole Publishing Company.

[111] Streeter, V.L., Wylie, E.B., and Bedford, K.W., *Fluid mechanics*. Ninth edition ed. 1998, Boston, Massachusetts: McGraw-Hill.

[112] Tang, C.J., Patel, V.C., and Landweber, L., *Viscous effects on propagation and reflection of solitary waves in shallow channels.* Journal of Computational Physics, 1990. **88**: p. 86-113.

[113] Tennekes, H. and Lumley, J.L., *A First Course in Turbulence* 1972: The MIT Press 300.

[114] Thom, A., *The Flow Past Circular Cylinders at Low Speeds, Series A, Containing Papers of a Mathematical and Physical Character*, in *Proceedings of the Royal Society of London* 1933. p. 651-669.

[115] Tiwari, S., Biswas, G., Prasad, P.L.N., and Basu, S., *Numerical prediction of flow and heat transfer in a rectangular channel with a built-in circular tube.* Journal of Heat Transfer, 2003. **125**: p. 413-421.

[116] Tokyay, T. and Constantinescu, S.G. *Coherent structures in pump intake flows: A Large Eddy Simulation study*. in *XXXI*st *International Association Hydraulic Research Congress*. 2005. Seoul, Korea.

[117] Tokyay, T., and Constantinescu, S.G. *Large Eddy Simulation and Reynolds Averaged Navier Stokes Simulations of flow in a realistic pump intake: A validation study*. in *World Water and Environmental Resources Congress*. 2005. Alaska.

[118] Tomiyama, A., Celata, G.P., Hosokawa, S., and Yoshida., S., *Terminal velocity of single bubbles in surface tension force dominant regime.* International Journal of Multiphase Flow, 2002. **28**: p. 1497-1519.

[119] Van de Sande, E. and Smith, J.M., *Surface Entrainment of Air by High Velocity Water Jets.* Chem. Eng. Sci, 1973. **28**: p. 1161-1168.

[120] Van Leer, B., *Towards the ultimate conservative difference scheme. V. A Second-order sequel to Godunov's method.* Journal of Computational Physics, 1979. **32**: p. pp 101-136.

[121] Vreman, A.W., *An Eddy-Viscosity Subgrid-scale Model for Turbulent shear flow: Algebraic theory and applications.* Physics of Fluids, 2004. **16**(10): p. 3670-3681.

[122] Wallis, G.B., *One-dimensional two-phase flow*. 1969, New York: McGraw-Hill.

[123] Wang, M. and Moin, P., *Dynamic wall modeling for large-eddy simulation of complex turbulent flows.* Physics of Fluids, 2002. **14**(7): p. 2043-2051.

[124] Xu, W., Wang, W., Yang, Y., and Chen, J., *Numerical modeling of the water-air two-phase jet into plunge pool.* Journal of Hydrodynamics, 1999. **11**(3): p. 1-5.

[125] Xu, W., Liao H., Yang, Y., and Wu, C., *Turbulent flow and energy dissipation in plunge pool of high arch dam.* Journal of Hydraulic Research, 2002. **40**(4): p. 471-476.

[126] Yang, S. and Lim, S., *Mechanism of Energy Transportation and turbulent flow in a 3D channel.* Journal of Hydraulic Engineering ASCE, 1997. **123**(8): p. 684-692.

[127] Yost, S.A., *Three-dimensional non-hydrostatic modeling of free surface turbulent flows and transport of cohesive sediment*, in *Department of Civil & Environmental Engineering*. 1995, University of Michigan: Ann Arbor, Michigan.

[128] Yost, S.A. and Rao, P., *A moving boundary approach for one-dimensional free surface flows* Advances in Water Resources, 2000. **23**(4): p. 373-382.

[129] Zang, Y., Street, R.L., and Koseff, J.R., *A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows.* Physics of Fluids A, 1993. **5**(12): p. 3186-3196.

[130] Zarrati, A.R., *Mathematical modeling of air-water mixtures in open channels.* Journal of Hydraulic Research/De Recherches Hydrauliques, 1994. **32**(5): p. 707-718.

[131] Zhao, H. and Voke, P.R., *A Dynamic Subgrid Scale Model For Low Reynolds-Number Channel Flow* International Journal For numerical methods in Fluids, 1996. **23**: p. 19-27.

VITA

Tien Mun Yee was born on July 12th 1978 in Kuala Lumpur, Malaysia. He received his Bachelors of Science (BSCE) and Masters of Science (MSCE) from the University of Kentucky. Tien Mun has attained several honors throughout his academic career, including Chi Epsilon Honor Society and Golden Key Honor Society. Tien Mun is also a member of the American Society of Civil Engineers and Kentucky Society of Professional Engineers.

Publications:

Yee, T.M., Yost, S.A., and Bradford, S.F. *Development of a 3-Dimensional Navie-Stokes equation model to solve for plunging water phenomenon*. [abstract] in *The 7th Int. Conf. on Hydroscience and Engineering (ICHE-2006)*. 2006. Philadelphia, USA: Advance in Hydro Science and Engineering.

Yee, T.M., Yost, S.A., and Bradford, S.F., *A novel approach to air bubble transport modeling using the scalar transport equation*, in *18th ASCE Engineering Mechanics Division Conference 2007*. 2007: Virginia, USA

Awards:

Recipient of Kentucky Water Resources Research Institute (KWRRI) funding (2007)
Awarded the Raymond Fellowship (2004-2005)

Professional Activities:

Graduate Student Congress representative for department of Civil Engineering (2006)
President of Graduate Forum group (2003)
President of KSPE University of Kentucky Chapter (2002)
Member of Coasts, Oceans, Ports, and Rivers Institute (COPRI)
Member of ASCE